Sources Probabilistes: des séquences aux systèmes

Jérémie Bourdon

l'université nantes angers le mans
PÔLE DE RECHERCHE ET D'ENSEIGNEMENT SUPÉRIEUR

UNIVERSITÉ DE NANTES

Habilitation à Diriger des Recherches

Jérémie Bourdon

Discipline : Informatique
Spécialité : Informatique
Laboratoire : Laboratoire d'informatique de Nantes-Atlantique (LINA)

Soutenue le 5 décembre 2012

École doctorale : 503 (STIM)
n° :

Sources probabilistes des séquences aux systèmes

JURY

Rapporteurs :
M^me^ Frédérique BASSINO, Professeure, Université Paris XIII
M. Alexander BOCKMAYR, Professeur, Freie Universität Berlin
M^me^ Hélène TOUZET, Directrice de recherche, CNRS Lille

Examinateurs :
M^me^ Anne SIEGEL, Directrice de recherche, CNRS Rennes
M^me^ Mireille RÉGNIER, Directrice de recherche, INRIA Saclay
M. Olivier ROUX, Professeur, Ecole Centrale de Nantes
M. Pierre COINTE, Professeur, Ecole des Mines de Nantes
M. Colin DE LA HIGUERA, Professeur, Université de Nantes
M. Gurvan MICHEL, Directeur de Recherche, Station Biologique de Roscoff
M^me^ Brigitte VALLÉE, Directrice de recherche, CNRS Caen

Remerciements

La liste pourrait être très longue et je risque d'en oublier.

Je remercie Frédérique Bassino, Hélène Touzet et Alexander Bockmayr pour avoir accepté d'être rapporteur de cette habilitation : merci pour leur lecture attentive et leurs remarques constructives. Je tiens aussi à remercier chaleureusement l'ensemble des membres qui composent mon jury d'habilitation : Mireille Régnier, Anne Siegel, Brigite Vallée, Pierre Cointe, Colin de la Higuera, Gurvan Michel, Olivier Roux.

Beaucoup de gens m'ont aidé et encouragé, le les remercie globalement : en premier lieu les membres de ComBi et plus généralement du LINA et du département d'informatique dont je partage le quotidien depuis 9 ans ; les Symbiotes qui m'ont si bien accueilli pendant deux années et chez qui j'ai toujours plaisir à retourner ; les membres de l'équipe MeForBio de l'IRCCyN avec lesquels je mène une collaboration continue depuis de nombreuses années ; Mireille Régnier qui m'a permis de découvrir la Russie au gré de collaborations fructueuses ; Rémi Houlgatte, co-encadrant de ma première doctorat Solenne Carat avec qui nos échanges s'éternisaient tellement qu'il était indispensable de les terminer devant un bon repas ; aux membres de l'INRA de Saint-Gilles qui m'ont fait découvrir les vaches à hublot mais surtout avec lesquels il est toujours agréable de travailler ; aux membres de la station biologique de Roscoff qui m'ont fait découvrir les oursins (sans hublot) mais aussi avec lesquels il est agréable de travailler ; les membres de la plateforme BiRD pour leur dynamisme.

Je remercie aussi tout particulièrement tout le groupe ALEA : en faire partie est la garantie d'avoir un soutien sans faille. Un grand merci aux Caennais qui m'ont permis de comprendre que le monde de la recherche est une grande famille dont je fais partie.

Je remercie enfin tout ceux qui m'ont aidé dans la poursuite de mes travaux, depuis le début de ma recherche jusqu'à aujourd'hui. Mener un travail interdisciplinaire n'est pas chose facile et je suis sûr de leurs encouragements, leur soutien et leurs collaborations sont pour beaucoup dans les résultats que je présente.

Je remercie ma famille, parents et beaux parents, frères soeurs, beaux-frères et belles-soeurs, je remercie mes amis qui par leurs encouragement ont contribué à mes travaux.

Je terminerai par les personnes qui me sont le plus chères. Je remercie Charlotte pour son ÉNORME soutien tout au long de ces années. Je termine par remercier notre si mignonne petite fille Justine.

Table des matières

Table des figures

1

Introduction et liste des travaux présentés

Avant propos

Ce mémoire est un recueil et une synthèse de plusieurs travaux en analyse en moyenne d'algorithmes et en bioinformatique. Il a été conçu pour être une grille de lecture permettant suivre ces travaux, allant de l'étude de problèmes sur les séquences à l'étude de systèmes biologiques, en gardant un fil conducteur fort : quelles que soient les applications, l'objet d'étude central est une source probabiliste qui produit des mots. Je présente donc des travaux allant de l'étude de séquences (avec des applications bioinformatiques qui se concrétisent par la mise au point d'algorithmes dédiés de recherche de motifs et la définition de tests statistiques) à l'étude de systèmes biologiques (avec des développements qui ont été appliqués, en collaboration étroite avec des équipes de biologistes, à des modèles biologiques réels).

1.1 Organisation du mémoire

Dans la première partie du manuscript, je présente les modèles mathématiques qui permettent de modéliser les sources de mots. Des outils classiques en analyse en moyenne d'algorithmes comme les séries génératrices et quelques théorèmes de transfert (des propriétés analytiques des séries génératrices vers leurs propriétés asymptotique) sont également exposés dans cette partie.

La seconde partie du manuscript se focalise sur les propriétés des objets émis par la source. Ici, on considère les mots et leur propriétés au travers de trois problématiques : comment tester si deux mots aléatoires sont similaires, en utilisant des fonctions de score complexes ; comment tester si un motif complexe est surreprésenté dans une séquence réelle ; quelle est la complexité spaciale de structures

de données qui permettent de stocker des séquences. Cette dernière question est fortement liée à la complexité des algorithmes de recherche et de découverte de motifs dans une séquence, les structures d'index étant au coeur de ces algorithmes.

La troisième partie adopte un point de vue assez différent. Plutôt que d'étudier les propriétés des objets émis, il est aussi judicieux d'étudier les propriétés des processus qui produisent ces séquences. Ceci trouve des applications naturelles en biologie des systèmes. Dans ce cadre, les processus complexes produisent des séquences qui peuvent être vues comme des traces laissées par le système sur son environnement, comme le sont les évolutions des états du système biologique par exemple. Il est généralement possible d'avoir des observations (plus ou moins fidèles) de ces séquences. Par contre, le processus lui-même n'est pas observable. Il n'est ainsi que très partiellement connu (voire totalement inconnu) et doit être reconstruit et étudié via le prisme de ces observations parcellaires. Le but de cette étude, un des buts premiers de la biologie des systèmes, est de comprendre le fonctionnement de ce système, d'en extraire des propriétés, de comprendre comment le perturber pour produire des séquences différentes. L'écho est donc naturel entre l'étude des propriétés de sources de mots et l'étude de systèmes biologiques. Enfin, une application à de vraies données biologiques est présentée. Ce travail représente une des rares applications concrètes des systèmes dynamiques à un problème biologique réel.

1.1.1 Séquences et ensemble de séquences.

La définition mathématique d'une séquence est relativement simple. Il suffit de fixer un alphabet qui sera noté Σ dans la suite et de considérer les objets $w \in \Sigma^\star$ qui sont des suites de longueurs finies et composées d'éléments de Σ. C'est un élément de base de nombreuses études. Selon la discipline, on l'appelera chaîne, mot, texte, séquence,.... En théorie de l'information, on considèrera souvent des suites de bits (*i.e.,*des mots sur l'alphabet $\{0, 1\}$) pour coder l'information. En traitement des langues naturelles, les mots, les phrases et les textes sont des séquences. En arithmétique, on s'intéressera aussi à des mots, par exemple les suites de quotients dans des développement en fractions continues, sur l'alphabet infini $\mathbb{Z}$. En bioinformatique, les mots sont des séquences biologiques (séquences ADN, ARN, protéiques,...). Bien évidemment, certains traitements concernent aussi des ensemble de chaînes que l'on appelle alors langages. Dans la suite, je fournis des éléments concernant les traitements qui peuvent être réalisés sur les chaînes ou les ensembles de chaînes comme par exemple la comparaison de deux chaînes, la recherche d'un langage motif dans une chaîne, ou le stockage d'un ensemble de chaînes.

Motivations biologiques

De nombreuses études et résultats biologiques reposent sur l'analyse de séquences dites biologiques (cf. [1] pour une référence). On parle bien de séquences biologiques, au pluriel, car elle peuvent revêtir plusieurs formes assez distinctes selon l'aspect biologique d'intérêt. La figure 1.1 présente le paradigme de base de la biologie moléculaire. Même si ce paradigme est remis en question, le cœur du problème est bien celui-ci : quelles informations sur le fonctionnement d'un organisme

peuvent être tirées de ces différentes séquences biologiques et surtout, comment obtenir ces informations.

Parmi les applications les plus populaires, on retrouve l'identification de gènes et de modules fonctionnels dans les séquences nouvellement séquencées (par recherche de motifs ou de similarité entre une séquence connue). Lorsque la séquence est la séquence d'une protéine, les applications sont également très nombreuses. Parmi elles, on peut noter l'annotation fonctionnelle ou la recherche de caractéristiques de familles de protéines. Deux problèmes informatiques liés à l'étude des séquences ont fait l'objet d'un intérêt très fort. La comparaison de séquences et la recherche/découverte de motifs sont en effet des outils de bases du bio-informaticien, lui permettant par exemple de prédire des interactions entre protéines [2, 3] ou des interactions entre gènes [4]. De telles identifications sont cruciales en biologie des système, notamment dans la phase de construction des réseaux d'interaction.

But poursuivi

Une des questions récurrentes lorsque l'on considère les problèmes de découverte de motifs ou de recherche de similarités consiste à se demander si le choix fait par l'algorithme est le bon. Autrement dit, pour la découverte de motifs, il s'agit de tester si le motif en question, dont on pense qu'il est important, est bien "statistiquement" un mot à prendre en compte. Une revue relativement complète sur ce sujet peut-être trouvée dans [5]. Pour répondre à ces questions, il faut définir des critères de décision d'ordre statistique. C'est un des buts que j'ai poursuivi au cours de mes recherches. Plus précisément, j'ai cherché à définir des modèles probabilistes qui permettent de se rapprocher le plus possible des caractéristiques des séquences biologiques réelles. J'ai aussi cherché à étudier des problèmes trouvant des applications directement pour le développement des algorithmes. Cette démarche a notamment été poursuivie pour étudier des scores de similarités entre séquences [6]. Cette étude a permis par exemple d'améliorer un algorithme de découverte de séquences biologiques. Pour être au plus près des motifs biologiques (de type Prosite par exemple), j'ai également étudié de manière précise le nombre d'occurrences d'expression régulières [7] et développé le concept de motif généralisé, une généralisation de la recherche de sous-séquences [8]. Enfin, parce que les structures d'index jouent un rôle important dans les implémentations d'algorithmes de découverte de motifs, j'ai proposé des résultats autour d'une structure d'index, l'oracle des facteurs, qui possède l'intérêt, au moins philosophique, d'être une structure d'index approchée et non exacte [9]. En effet, c'est à mon avis via des structures d'index de ce type (*i.e.,*qui stockent une approximation d'un ensemble de mots, c'est aussi le cas des filtres de Bloom) que les problèmes de comptage/assemblage/indexation de gros ensembles de séquences pourront être traités, avec des applications évidentes pour gérer les données de séquençage à très haut débit.

1.1.2 Systèmes biologiques

Les systèmes biologiques sont par nature des systèmes complexes, qui font intervenir une large variété d'acteurs à diverses échelles de taille (cellule, tissu, organisme, population) et de temps. La biologie des systèmes est la science qui s'em-

ploie à développer des modèles mathématiques, informatiques permettant de représenter ces modèles vivants, les étudier et en tirer des hypothèses dites biologiques, qui doivent être vérifiées expérimentalement. Parce que les aspects de modélisation sont au centre de la discipline et que la conception du modèle est multi-facette, l'étude des systèmes biologiques est une discipline au carrefour de la biologie, des mathématiques, de l'informatique, de la physique, de la chimie,...

Jusqu'à présent, des formalismes de modélisation ont été proposés pour étudier une échelle fixée. Très peu de tentatives ont été proposées pour modéliser plusieurs échelles (d'espace ou de temps). C'est un des challenges futurs de la discipline que de réconcilier plusieurs échelles du vivant. Ainsi, j'ai poursuivie cette piste en me focalisant particulièrement, dans un premier temps sur les réseaux de régulation de gènes.

Motivations biologiques

L'expression d'un gène – ou synthèse protéique – comporte les deux étapes de la transcription et de la traduction (cf Figure 1.1 pour une vision idéalisée des phénomènes biologiques) : grâce à l'intervention d'une molécule d'ARN polymérase, le segment d'ADN correspondant au gène est transcrit en une molécule d'ARN messager ; cette molécule d'ARN messager est ensuite traduite en la protéine codée par le gène. L'expression génique est un processus complexe susceptible de régulations à divers instants de cette synthèse. Une régulation peut consister en une activation ou une inhibition. Comme les protéines remplissant cette fonction régulatrice sont produites par des gènes, il y a lieu de parler de réseau de régulation génique, structuré par un réseau d'interactions entre molécules d'ADN, d'ARN, de protéines et d'autres molécules de plus petite taille, les métabolites.

La production à grande échelle de données issues de l'analyse du transcriptome, du protéome et du métabolome a fortement accéléré l'essor de travaux destinés à modéliser la dynamique du vivant. Une modélisation appropriée rend possible des simulations, dont sont attendues la vérification de propriétés, l'identification des étapes sensibles ou au contraire robustes de la dynamique des systèmes analysés. Lorsque le système concerné est un réseau de régulation de gènes, tout travail prospectif peut s'appuyer sur un socle à trois composantes, avec retours d'expériences et affinages successifs du modèle : (i) apprentissage du modèle par observations massivement parallèles, spatio-temporelles, du niveau d'expression des gènes (puces à ADN, à oligonucléotides...), (ii) modélisation des réactions ou interférences entre molécules, (iii) simulation. Pour vérifier certaines propriétés du réseau, une alternative à la simulation peut être exploitée : la vérification formelle de propriétés.

Parmi les questions auxquelles biologistes et biochimistes peuvent souhaiter voir apporter une réponse figurent les suivantes : Quels sont les états par lesquels peut transiter le système étudié ? Quels sont les états stables, instables ? En combien de temps en moyenne les atteint-on à partir d'un état initial donné ? Quels sont les divers chemins possibles, à partir d'un état initial donné ? A partir d'un état, peut-on atteindre un autre état ? Si oui, en combien de temps en moyenne ? Peut-on identifier des classes de comportements du système, reliées à des caractéristiques données du système ? Par exemple, lorsque l'état initial appartient à telle classe, il est certain

que le système passe par tel état, ou encore lorsque l'état initial appartient à telle classe, il est certain que le système aboutit à tel état.

Une modélisation au moyen d'un graphe permet déjà d'inférer certaines informations sur le réseau de régulation modélisé. Par exemple, l'absence de chemin reliant deux gènes qu'on sait par ailleurs être en interférence met en évidence un défaut du modèle (interactions manquantes). L' identification de chemins multiples reliant deux gènes informe sur les redondances du système biologique étudié. La présence de cycles peut révéler des boucles de rétro-action (*feedback*).

Les questions précédentes correspondent à des modèles simples. Dès lors que l'on introduit des paramètres destinés à une description plus fine du système biologique, par exemple la notion de niveau d'expression d'un gène (passage du qualitatif au quantitatif), celle de délai de transition d'un état à un autre, ou encore la notion d'alternative, associée à une probabilité de choix d'alternative, d'autres questions apparaissent. Par exemple, on peut se poser la question d'identifier, si possible, les domaines de valeurs pour les paramètres du système qui induisent des comportements similaires.

Il est important de pouvoir vérifier des propriétés sur un système biologique sans recourir à la simulation d'un nombre de cas d'autant plus élevé que le modèle est complexe. Ainsi apparaît-il essentiel de disposer d'une part d'un modèle décrivant de manière suffisamment fine la réalité biologique, et d'être capable d'autre part de traduire ce modèle à l'aide d'un formalisme, basé sur des automates ou des processus de type markovien. Le formalisme est destiné à permettre la vérification de propriétés, au moyen d'un raisonnement basé sur une théorie. Parmi ce dernier type de raisonnements, figure le raisonnement symbolique, sur lequel s'appuient des outils de vérification de propriétés dans les systèmes.

But poursuivi

Mes travaux visent à considérer les systèmes biologiques comme des processus stochastiques sophistiqués et à en comprendre le fonctionnement. L'utilisation de modèles probabilistes en biologie des systèmes n'est pas nouvelle. Il a été notamment montré que des chaînes de Markov pouvaient être de bons modèles de départ pour modéliser certains points du vivant [10]. D'autres modèles ont obtenu également de vrais succès pour représenter la dynamique des réseaux de gènes. Ces modèles, comme les Probabilistic Boolean Networks (PBN) [11], ou encore les Random Boolean Networks [12], introduisent une composante stochastique aux réseaux booléens qui permettent de représenter les graphes d'états de réseaux de régulation de gènes.

Dans mes travaux, je me suis dans un premier temps attaché à enrichir l'expressivité des réseaux de gènes. Cela a consisté à ajouter une composante temporelle dans la description de la dynamique des réseaux de gènes. Par exemple, dans les PBN, l'évolution est considérée comme purement synchrone, décrite par une fonction booléenne. Dans [13], nous avons montré que des stratégies complexes d'évolution peuvent être appliquées aux PBN. Ces stratégies incluent des priorités de gènes les uns sur les autres (dues par exemple à différentes affinités biochimiques dans les processus d'activation) et des coopérations entre gènes (qui font que leurs

évolutions sont couplées). Ensuite, nous avons montré dans [14] que les délais de réaction jouaient un rôle dans l'évolution de la dynamique. En introduisant les délais en tant que paramètres dans le graphe d'états, nous avons pu montrer que certains attracteurs n'étaient possibles que sous certaines hypothèses (contraintes de supériorité) sur les délais. Que certains cycles, proposés par d'autres méthodes étaient finalement impossible lorsque ces contraintes de délais sont présentes.

Dans un second temps, je me suis attaché à ajouter une composante quantitative aux réseaux discrets. L'idée derrière cela part de ce constat : il n'existe quasiment aucune observation biologique permettant de valider les résultats obtenus sur les modèles discrets (qui décrivent la dynamique d'un individu). Pourtant, il existe de nombreuses observations, comme tout simplement des évolutions de concentration de protéines, qui sont des conséquences de la dynamique du réseau 'ìndividuel" mais observées au niveau d'une "population d'individus". Nous avons montré qu'en ajoutant une composante quantitative aux réseaux discrets (sous forme de poids d'évènements biologiques) et en étudiant des propriétés asymptotiques moyennes de ces quantités (pour se ramener à l'échelle des populations), de nombreuses observations peuvent alors être utilisées pour valider le réseau et apporter des informations complémentaires. Ces résultats sont synthétisés dans [15] et [16].

1.2 Travaux présentés

1.2.1 Problèmes sur les mots et les ensembles de mots : bioinformatique des séquences et génomique

Les articles de ma bibliographie qui se rattachent à cette thématique sont les suivants. Mon propos se reposera précisément sur les articles marqués d'une étoile $\star$.

- $\star$ Statistical Properties of Factor Oracles, Bourdon, J. and Rusu, In proceedings of CPM 2009, LNCS 5577, 326-338, 2009, version étendue publiée dans Journal of Discrete Algorithms, 2010, 9 (2011), pp. 59-66
- A parallel scheme for comparing transcription factor binding sites matrices, Solenne Carat, Rémi Houlgatte and Jérémie Bourdon, Journal of Bioinformatics and Computational Biology 8(3), pp. 18 (2010).
- A statistical method for PWM clustering, Solenne Carat ; Rémi Houlgatte ; Jérémie Bourdon, *Proceedings of Moscow Conference on Computational Molecular Biology, Jul 2009, Moscou, Russian Federation. pp. 59-60*
- Combinations of cytochrome p450 gene polymorphisms enhancing the risk for sporadic colorectal cancer related to red meat consumption. Sébastien Küry, Bruno Buecher, Sébastien Robiou du Pont, Catherine Scoul, Véronique Sébille, Hélène Colman, Claire Le Houérou, Tanguy Le Neel, Jérémie Bourdon, Roger Faroux, Jean Ollivry, Bernard Lafraise, Louis-Dominique Chupin, and Stéphane Bézieau. Cancer Epidemiology Biomarkers and Prevention, 16 :1460–1467, 2007.
- $\star$ Statistical Properties of Similarity Score Functions, Bourdon, J. and Mancheron, A., In Proceedings of the 4th Colloquium on Mathematics and Computer Science. Algorithms, Trees, Combinatorics and Probabilities, Discrete

Mathematics and Theoretical Computer Science (DMTCS), pages 129-140, (2006)

⋆ Pattern Matching Statistics on Correlated sources, Bourdon, J and Vallée, B., Proceedings of LATIN'06, LNCS 3887, pp 224-237. (2006)

– Generalized Pattern Matching Statistics, Jérémie Bourdon, Brigitte Vallée, *Proceedings of colloquium on Mathematics and Computer Science : Algorithms and Trees*, Birkhauser, Trends in Mathematics, pp. 249-265 (2002)

– Size and Path Length in Patricia Tries : Dynamical Sources Context, Jérémie Bourdon, *Random Structures and Algorithms*, 2001, 3-4 (19), pp. 289-315

1.2.2 Comprendre, analyser, manipuler une source de mots : bioinformatique des systèmes.

Voici les articles concernés par cette thématique. Mon propos concernera principalement les articles marqués d'une étoile ⋆.

⋆ Integrating quantitative knowledge into a qualitative gene regulatory network, Jérémie Bourdon ; Damien Eveillard ; Anne Siegel, PLoS Computational Biology, 2011, 7 (9), pp. e1002157.

– Probabilistic Approaches for Investigating Biological Networks, Jérémie Bourdon ; Damien Eveillard, *Algorithms in Computational Molecular Biology : Techniques, Approaches and Applications,* Mourad Elloumi, Albert Y. Zomaya (eds), Wiley, pp. 1066, Janvier 2011

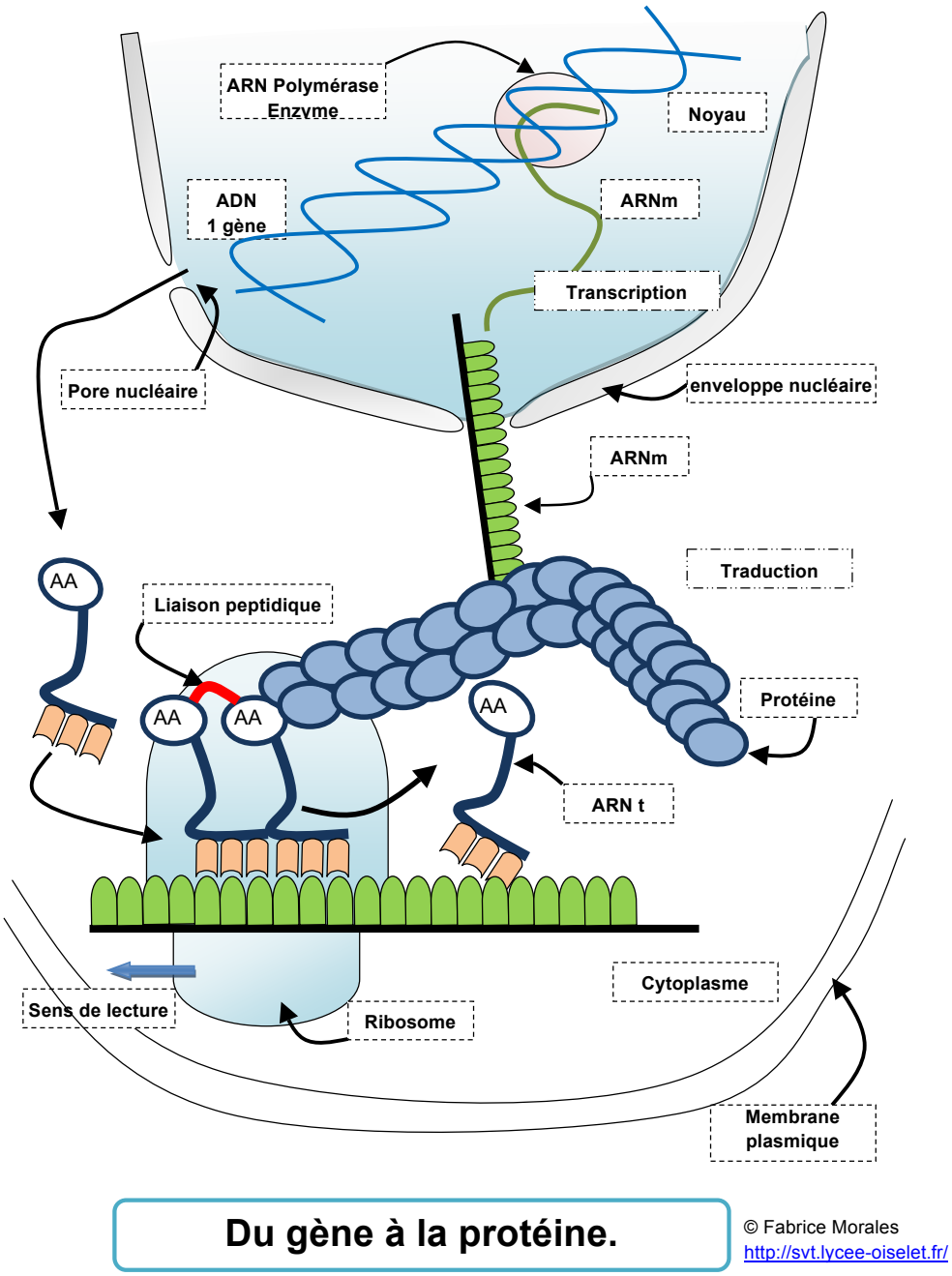
⋆ Toward systems biology in brown algae to explore acclimation and adaptation to the shore environment, Thierry Tonon ; Damien Eveillard ; Sylvain Prigent ; Jérémie Bourdon ; Philippe Potin ; Catherine Boyen ; Anne Siegel, *OMICS*, 2011, 15 (12), pp. 883-92

⋆ Complex update strategies for Probabilistic Boolean Networks, Merle, T. and Bourdon, J., Proceedings of WCSB 2010, Luxembourg, juin 2010.

– A generic stoichiometric model to analyse the metabolic flexibility of the mammary gland in lactating dairy cows, Sophie Lemosquet ; Oumarou Abdou-Arbi ; Anne Siegel ; Jocelyne Guinard-Flament ; Jaap Van Milgen ; Jérémie Bourdon, *Modelling nutrient digestion and utilization in farm animals*, D. Sauvant, J. Van Milgen, P. Faverdin and N. Friggens (eds) Wageningen Academic Publishers, 2010, 978-90-8686-156-9

⋆ Temporal constraints of a gene regulatory network : Refining a qualitative simulation, Ahmad, J. ; Bourdon, J. ; Eveillard, D. ; Fromentin, J. ; Roux, O. et Sinoquet, C., Biosystems 98(3), p. 149-159, (2009)

FIGURE 1.1 – Le paradigme de base en biologie moléculaire. Une séquence ADN est transcrite en une séquence ARNm qui est elle-même traduite en une séquence protéique.



2

Présentation des outils mathématiques utilisés

L'analyse en moyenne des algorithmes vise à étudier le comportement "moyen" de l'algorithme, c'est à dire, en un sens informel son comportement sur ses entrées les plus typiques. Quand il s'agit d'algorithmes de texte, les différents coûts sont exprimés dans un contexte où les mots en entrée (ou en sortie) interviennent par le biais d'objets caractéristiques liés à la source aléatoire qui a émis ces mots (la probabilité d'émettre un mot (fini), une composition d'homographies ou d'opérateurs,...). Ainsi, il est primordial dans ces études de définir de manière convenable ce qu'est un processus aléatoire de création de mots. La suite montre qu'il existe bien des définitions de ce que peut-être une source aléatoire. Nous terminons la première partie en décrivant un modèle général, et en quelquesorte unificateur, de définition de source aléatoire de mots. Ensuite, nous introduisons quelques outils qui permettent d'étudier les problèmes sur les mots dans un cadre (de sources) classique(s), les séries génératrices, puis nous étendons cette notion à un cadre de sources plus général, via des opérateurs générateurs. Enfin, nous donnons quelques résultats, des théorèmes de transfert, qui permettent de passer des propriétés analytiques des séries génératrices (principalement le type et la localisation des singularités de ces séries, lorsqu'elles sont vues comme des fonctions sur le plan complexe) et l'asymptotique des coefficients de la série génératrice, eux mêmes liés à une caractéristique sur les mots (par exemple, le nombre moyen d'occurrences d'un motif dans un mot de taille n, quand n est grand). Ces modèles et outils sont centraux dans les résultats exposés par la suite. Les ouvrages à consulter pour une présentation exhaustive de ces sujets sont [17, 18] pour les aspects liés aux sources de mots et [19] pour les théorèmes de transfert.

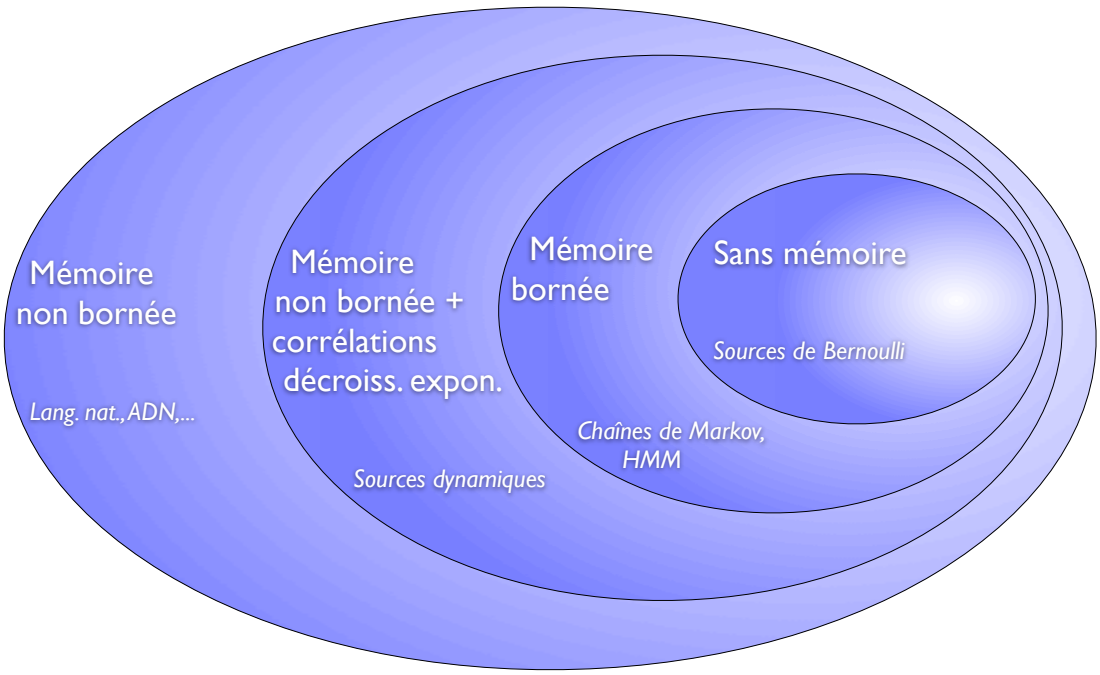
2.1 Modèles probabilistes pour produire des mots.

La plupart des résultats de ce mémoire sont liés de près ou de loin à des résultats sur des séquences. L'objet d'étude est la plupart du temps une variable aléatoire construite en tant que mesure (calculée de manière déterministe) sur une séquence aléatoire. Ainsi, la question de trouver un modèle réaliste pour l'objet d'étude se ramène à trouver un modèle réaliste de génération de séquences aléatoires. Dans ce mémoire, nous parlerons de sources aléatoires de mots. Ces sources aléatoires de mots permettent en fait de définir une suite de mesures de probabilité sur les espaces Σ^n, ensembles des mots de longueur $n \geq 0$ sur l'alphabet Σ (fini ou dénombrable). Autrement dit, une probabilité p_w est associée à chaque mot $w \in \Sigma^*$ telle que

$$\sum_{w \in \Sigma^n} p_w = 1.$$

Ici, le nombre p_w est la probabilité qu'un mot (infini) commence par le préfixe (fini) w. La figure 2.1 présente quelques types classiques de sources de mots que l'on distingue par le degré de corrélation entre symboles successifs (plus la corrélation s'étend, plus la source est difficile à étudier mais en contrepartie, plus elle se rapproche de la réalité).

FIGURE 2.1 – Les principales classes de sources de mots probabilistes.



Une source est un processus discret qui produit des symboles issus d'un alphabet Σ (fini ou dénombrable). Chaque symbole émis peut ou non dépendre des symboles émis avant lui. Dans le cas le plus général, la probabilité d'émettre un symbole $m \in \Sigma$ est conditionnée par le préfixe $w \in \Sigma^\star$ (l'ensemble des symboles) émis

avant lui et est la probabilité conditionnelle

$$p_{[m|w]} := \frac{p_{w\cdot m}}{p_w},$$

où p_w est la probabilité du préfixe w (*i.e.*, la probabilité qu'un mot infini émis par la source commence par w) et vérifie $\sum_{|w|=k} p_w = 1$, $k \geq 0$.

Pour la plupart des problèmes étudiés ultérieurement, l'analyse comporte essentiellement deux parties : une partie algébrique correspondant essentiellement à une description du problème pour laquelle cette modélisation de la source suffit et une partie analytique correspondant à l'étude des séries de probabilités associées au problème où des hypothèses supplémentaires sur les probabilités des préfixes doivent être ajoutée pour mener à bien cette analyse.

2.1.1 Sources classiques

Dans le cas des sources classiques, la probabilité n'est conditionnée que par au plus une partie du préfixe. Pour les sources sans mémoire, la probabilité d'émettre le symbole est fixe au cours du temps et ne dépend par du tout du préfixe. Pour une chaîne de Markov d'ordre k, la probabilité est conditionnée par les k derniers symboles du préfixe.

Sources sans mémoire

Ces sources sont certainement les plus simples et les plus étudiées en théorie de l'information. Son principe est le suivant : on se donne un alphabet Σ et les fréquences d'apparition des symboles $\{p_m\}_{m\in\Sigma}$. Ensuite, le symbole m est émis indépendamment de tout autre symbole avec la probabilité p_m.

Lorsque les symboles sont tous émis avec la même probabilité $1/|\Sigma|$, la source est dite *non biaisée*. Le mot

```
yvryvndfckfbavduxhzopy akoomyr
```

est un exemple de mot produit par une source non biaisée de probabilité $1/27$ (les symboles sont ici les 26 lettres de l'alphabet et l'espace).

Les sources sont en général destinées à modéliser le plus fidèlement possible des sources "naturelles". Si l'on désire une source permettant de modéliser la langue française, la première étape consiste à tenir compte des fréquences réelles d'apparition des différents symboles, chaque symbole étant ensuite émis indépendamment par une source sans mémoire. Pour obtenir une approximation de ces fréquences, on peut par exemple utiliser un texte comme échantillon (ici le texte échantillon est "Les mémoires d'outre-tombe" de Chateaubriant). Un exemple de texte obtenu par une telle source

```
jetrrneaeaep s mtunci potme eloshfn rsi aa e auu mbo
```

se rapproche déjà plus, d'un point de vue syntaxique d'un texte en français.

Chaînes de Markov

L'étape suivante consiste à tenir compte du symbole qui vient d'être émis. La source correspondant s'appelle alors chaîne de Markov (d'ordre 1).

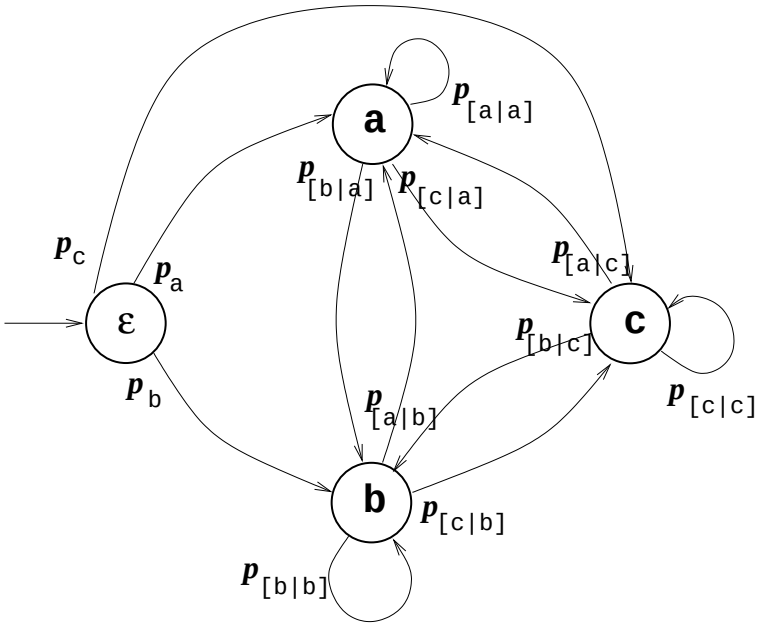
La suite de symboles `qu` est assez caractéristique de l'amélioration apportée par ce modèle. En effet, la suite de symbole `qu` apparaît environ 20 fois plus souvent dans un texte émis par une chaîne de Markov d'ordre 1 (construite à l'aide de probabilités issues d'un texte échantillon) que dans un texte émis par une source sans mémoire (relative au même échantillon). L'exemple de texte

```
de anochent ailanqute leserau ge pat desol sea d
```

émis par cette chaîne de Markov est déjà plus proche d'un texte réel. On y retrouve des découpages cohérent (il n'y a plus de suite de deux espaces) et quelques sous-mots caractéristiques apparaissent (chent, desol,...). Même si l'ensemble reste totalement dénué de sens, la bonne alternance entre les consonnes et les voyelles permet de "prononcer" les mots.

Une telle chaîne de Markov peut également être interprétée à l'aide d'un automate non déterministe. Les états de l'automate servent à mémoriser la dernière lettre émise et les transitions partant d'un état (relatif à une lettre ℓ) sont étiquetées par la lettre qui va être émise m et sont pondérées par les probabilités $p_{[m|\ell]}$.

FIGURE 2.2 – Un automate associé à une chaîne de Markov d'ordre 1



On peut encore complexifier la source et considérer que chaque symbole dépend des deux derniers symboles émis. Il s'agit alors d'une chaîne de Markov d'ordre 2. On obtient un texte ressemblant à

```
te de les je m abditimencieurs il le d ation mait etroi.
```

Des mots (encore courts) se dessinent alors.

A partir de l'ordre 6, des phrases quasiment cohérentes sont émises :

```
une consulte pour l evasion de la porte quinze cent
```

Entropie d'un langage

Des études pour quantifier la cohérence d'une source avec un langage naturel ont été menées (pour la langue anglaise) par Shannon [20] et reprisent par Welsh [21]. Ils ont ainsi montré que dans une suite aléatoire de lettres prises dans l'alphabet usuel, l'information portée par chacune des lettres est $\log_2(27) \simeq 4.75$ bits. En tenant compte des fréquences des différentes lettres, on obtient environ 4.029 bits pour l'anglais et 3.949 bits pour le Français sur le texte échantillon utilisé. On peut aussi tenir compte des fréquences des suites de deux symboles (encore appelées digrammes), l'information apportée par chaque symbole devient 3.318 bits pour l'anglais et 3.192 bits pour le français. Ces quantités d'information moyenne sont appelées *entropies* des sources aléatoires considérées. Pour une source sans mémoire $\mathcal{S}$ de probabilités $\{p_m\}_{m\in\Sigma}$, elle s'exprime sous la forme

$$H(\mathcal{S}) = \sum_{m\in\Sigma} p_m \log p_m.$$

Des expressions similaires peuvent être obtenues pour les chaînes de Markov d'ordre r quelconque. Ces expressions pour être calculées nécessitent le calcul des vecteurs propres d'une matrice (la matrice des transitions de la chaîne de Markov) de dimension $r+1$.

L'entropie $H(\mathcal{L})$ d'un langage naturel $\mathcal{L}$ peut également être définie comme étant la limite quand $r \to \infty$ des entropies des chaînes de Markov $\mathcal{S}_r$ d'ordre $r-1$,

$$H(\mathcal{L}) := \lim_{r\to\infty} H(\mathcal{S}_r).$$

Les études menées pour déterminer l'entropie des langues anglaise et française ont mené à des valeurs de l'ordre de $1,25$ bits. En moyenne, une lettre d'un texte ne porte donc pas beaucoup plus d'un bit d'information (ceci explique par exemple que les fichiers textes peuvent être compressés très efficacement). Cette valeur, si elle est comparée à l'entropie d'une source sans mémoire non biaisée, peut s'interpréter de la manière suivante : pour écrire un texte sensé, seuls 25% des lettres peuvent être choisies arbitrairement, les autres lettres étant imposées par les règles de structure du langage.

2.1.2 Sources dynamiques

Les sources dynamiques ont été introduites par Vallée [22]. Elles fournissent un cadre général qui permet de modéliser non seulement les sources classiques (sans mémoire ou chaîne de Markov) qui prennent en compte un nombre borné de symboles corrélés, mais aussi des sources plus complexes comme les développements en fractions continues pour lesquelles chaque symbole émis dépend de l'ensemble des symboles déjà produits.

Définition

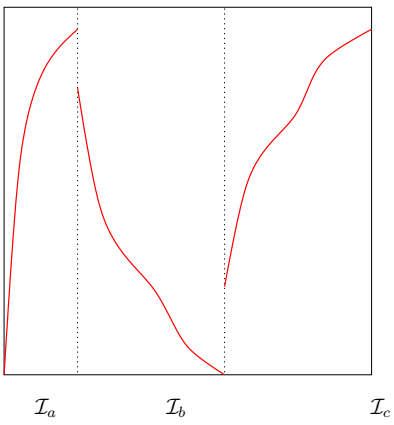
Une source dynamique est un processus probabiliste qui se décompose essentiellement en deux étapes. Premièrement, un réel x de l'intervalle $\mathcal{I} :=]a, b[$ (le plus souvent, $\mathcal{I} =]0, 1[$) est choisit aléatoirement selon une certaine fonction de densité f. Ensuite, ce réel est utilisé pour produire des mots infinis suivant un mécanisme totalement déterministe.

Définition 1 *Une source dynamique est définie par :*

- (i) *un alphabet Σ fini ou dénombrable,*
- (ii) *une partition topologique $(\mathcal{I}_m)_{m\in\Sigma}$ de l'intervalle $\mathcal{I}$, (*i.e., $\overline{\mathcal{I}} = \cup_{m\in\Sigma}\overline{\mathcal{I}}_m$*),*
- (iii) *une fonction de codage $\sigma : \mathcal{I} \to \Sigma$ constante et égale à m sur chaque intervalle $\mathcal{I}_m$,*
- (iv) *une fonction de décalage $T : \mathcal{I} \to \mathcal{I}$, telle que la restriction $T_m := T|_{\mathcal{I}_m}$ de T à l'intervalle $\mathcal{I}_m$ est une fonction monotone de $\mathcal{I}_m$ dans $\mathcal{J}_m := T_m(\mathcal{I}_m)$ et de classe $\mathcal{C}^2$.*

Ce mécanisme lorsqu'il est associé à une fonction de densité f sur $\mathcal{I}$, est appelé *source dynamique probabiliste*.

FIGURE 2.3 – Un exemple de source dynamique sur l'alphabet $\Sigma = \{a, b, c\}$.



Emission d'un mot à l'aide d'une source dynamique

L'*orbite* d'un réel x de $\mathcal{I}$ est la séquence de réel

$$O(x) := (x, Tx, T^2x, T^3x, \dots)$$

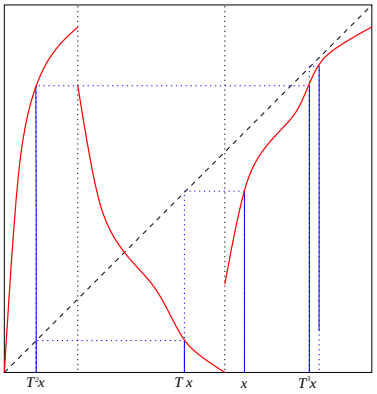
dont les composantes sont les réels T^ix. Si l'on applique la fonction de codage σ à cette séquence, on obtient donc le mot infini

$$M(x) := \sigma(O(x)) = (\sigma(x), \sigma(Tx), \sigma(T^2x), \sigma(T^3x), \dots).$$

Le processus pour produire un mot à l'aide d'une source dynamique probabiliste est alors le suivant :

– premièrement, on choisit un réel x de $\mathcal{I}$ selon la fonction de densité f,
– ensuite, le $i^{ème}$ symbole du mot émis est $\sigma(T^i x)$. Le mot $M(x)$ obtenu est appelé *mot associé au réel* x et est noté $M(x)$.

FIGURE 2.4 – Le mot émis par la source dynamique est $cbacc\ldots$.



Le figure 2.5 présente plusieurs exemples de sources dynamiques. Notamment, les sources classiques (sans mémoires et chaînes de Markov) sont des sources dynamiques pour lesquelles les branches sont affines (ou affines par morceau dans le cas des chaînes de Markov). La pente des branches étant égale à l'inverse de la probabilité d'émettre le symbole.

Il existe une forte relation entre les fonctions d'encodage et de décalage définies pour des réels et les fonctions $\underline{\sigma}$ et $\underline{T}$ qui retournent respectivement la première lettre et le premier suffixe d'un mot. En effet, les quatre fonctions σ, T, $\underline{\sigma}$ et $\underline{T}$ vérifient

$$\sigma(x) = \underline{\sigma}(M(x)), \qquad \sigma(Tx) = \underline{\sigma}(\underline{T}M(x)).$$

Remarque: Le mot infini $M(x)$ mémorise la trajectoire du réel x. En effet, si $M(x)$ admet comme préfixe $w := m_1 m_2 \ldots m_k$, la $(k+1)^{ème}$ composante de l'orbite $O(x)$ est le réel

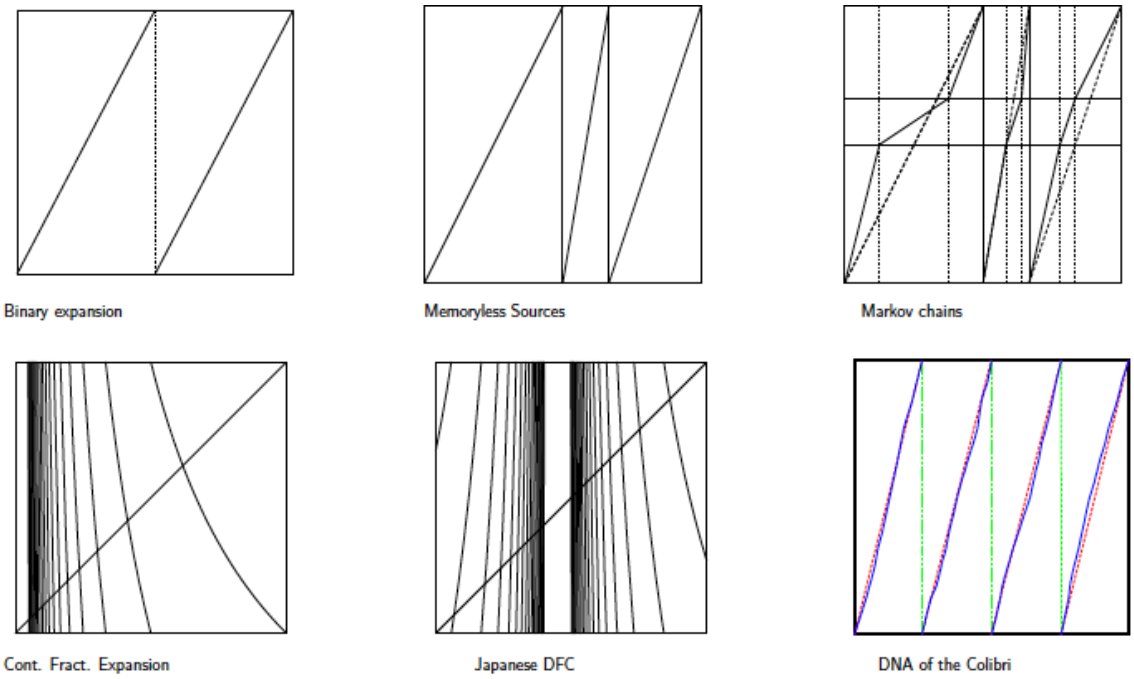
$$T^k x = T_{m_k} \circ \cdots \circ T_{m_2} \circ T_{m_1} x := T_w x. \tag{2.1}$$

On remarque aisément que l'ensemble des mots qui débutent par le même préfixe w sont issus d'un intervalle $\mathcal{I}_w$, appelé *intervalle fondamental* associé au préfixe w. La fonction $T^k|_{\mathcal{I}_w} = T_w$ est alors une bijection de $\mathcal{I}_w$ dans $\mathcal{J}_w := T_w(\mathcal{I}_w)$. Son inverse local est la fonction $h_w : \mathcal{J}_w \to \mathcal{I}_w$ égale, d'après (2.1), à $h_{m_1} \circ \cdots \circ h_{m_k}$.

Ainsi, la probabilité p_w de débuter par le préfixe w n'est autre que la mesure de l'intervalle fondamental $\mathcal{I}_w$,

$$p_w = \int_{\mathcal{I}_w} f(t)dt. \tag{2.2}$$

FIGURE 2.5 – Quelques exemples de sources dynamiques.



Bonnes sources dynamiques

Pour être utilisées en théorie de l'information, les sources dynamiques doivent vérifier un certain nombre d'hypothèses relativement faibles : elles opèrent sur un alphabet fini ; les branches des sources sont strictement expansives ; et la source est topologiquement mélangeante. Il faut noter que la condition de finitude de l'alphabet peut être levée au profit d'un alphabet dénombrable à condition d'ajouter des conditions de convergences qui rendent les preuves plus complexes. Nous n'aborderons pas ce cas. Voici ces trois propriétés décrites plus formellement.

(F) Σ est fini ;

(E) Il existe deux constantes C et D avec $D > 1$ telles que pour tout $m \in \Sigma$ et tout $x \in \mathcal{I}_m$, $D < |S'(x)] < C$;

(M) Pour toutes paires d'ensembles non vides (V, W), il existe $n_0 \geq 1$ tel que $T^{-n}V \cap W \neq \emptyset$ pour tout $n \geq n_0$

Lorsque qu'une source vérifie ces trois conditions, elle est dite "bonne" et son opérateur transformateur de densité $\mathbb{G}$, qui décrit l'évolution de la densité de probabilité après une itération de la fonction de shift T, et qui est défini (pour toute fonction f et tout point $x \in \mathcal{I}$) par

$$\mathbb{G}[f](x) := \sum_{m \in \Sigma} |(h_m)'(x)| f \circ h_m(x), \text{ où } h_m := T_m^{-1},$$

lorsqu'il s'applique à l'espace de Banach des fonctions à variations bornées $BV(\mathcal{I})$ muni de la norme $||f|| := \sup |f| + V(f)$, où$V(f)$ est la variation totale de f sur $\mathcal{I}$, possède une unique valeur propre dominante réelle (et égale à 1), séparée du reste

du spectre par un saut spectral, i.e., $\rho := \sup\{|\lambda| \ ; \quad \lambda \in \operatorname{Sp}\mathbb{G}, \lambda \neq 1\} < 1$. Ainsi, $\mathbb{G}$ se décompose en deux parties, $\mathbb{G} = \lambda\mathbb{P} + \mathbb{N}$, où $\mathbb{P}$ est la projection de $\mathbb{G}$ sur le sous espace propre dominant et $\mathbb{N}$, relatif au reste du spectre a un rayon spectral ρ, strictement inférieur à 1. L'opérateur $\mathbb{N}$ décrit les corrélations de la source. Une source dynamique qui admet une telle décomposition est ergodique et mélangeante (avec un taux exponentiel égal à ρ).

2.2 Analyse en moyenne, séries génératrices et autres

La série génératrice est un outil de choix en analyse d'algorithme. En effet, ces séries génératrices permettent d'encoder les caractéristiques principales dans un grand nombre de problèmes. En étudiant leur principales propriétés, il est possible de déduire les coûts asymptotiques des grandeurs qu'elles représentent.

Plus précisément, nous présentons ici deux "transferts" différents. Le premier transfert consiste à exprimer le problème en terme de séries génératrices formelles. Les opérations qui existent sur les structures se transfèrent (dans le cas des sources simples) à des opérations sur les séries génératrices. Un "dictionnaire" permet alors d'obtenir, pour ces séries, des expressions qui permettent de les étudier. Le but de la première partie du chapitre suivant consiste à obtenir un dictionnaire similaire dans le cas de sources plus générales.

Pour le second transfert, les séries génératrices formelles sont désormais considérées comme des séries génératrices complexes. Ces séries possèdent des singularités qui dictent le comportement asymptotique de leurs coefficients. Ce comportement est étudié par le biais de théorèmes qui transfèrent les comportements autour des singularités aux comportements asymptotiques.

2.2.1 Séries génératrices de mots

Les séries génératrices sur les mots sont des séries entières de probabilités. Nous définissons ici successivement les séries génératrices ordinaires et exponentielles (dans leurs formes uni-variées et multivariées) qui sont classiquement utilisées pour le dénombrement d'objets et l'étude de paramètres sur les mots produits par des sources uniformes ; puis les séries de probabilités utilisées pour étudier les problèmes sur les mots pour des sources générales.

Séries entière de probabilités

Lorsque l'on veut étudier certains paramètres dans le cas de sources de mots générales, il est indispensable d'utiliser des séries faisant intervenir les probabilités d'émission des symboles.

Tout d'abord, une source probabiliste (générale) $\mathcal{S}$ est un processus (discret) qui produit des suites (infinis) de symboles (les mots). Un mot infini commencera par le préfixe w avec une probabilité p_w telle que $\sum_{|w|=k} p_w = 1$, pour tout $k \geq 0$.

Nous définissons tout d'abord les séries génératrices de probabilités uni-variées. Elles sont définies pour une classe d'objets $\mathcal{A}$ et un paramètre sur cette classe souvent appelé *taille*. Les classes d'objets que nous considérons sont des ensembles

de mots ou des collections de mots (chaque mot peut apparaître plusieurs fois). La taille $|w|$, $w \in \mathcal{A}$ est la longueur du mot w. Les *séries de probabilités ordinaire* et *exponentielle* associée à l'ensemble (ou la collection, *i.e.,* un même mot peut apparaître plusieurs fois) $\mathcal{A}$, relative à la source probabiliste $\mathcal{S}$ sont définies par

$$F_{\mathcal{A}}(z) = \sum_{w\in\mathcal{A}} p_w z^{|w|} = \sum_{n\geq 0}\sum_{w\in\mathcal{A}_\backslash} p_w z^n, \qquad \widetilde{F}_{\mathcal{A}}(z) = \sum_{w\in\mathcal{A}} p_w \frac{z^{|w|}}{|w|!} = \sum_{n\geq 0}\sum_{w\in\mathcal{A}_\backslash} p_w \frac{z^n}{n!},$$

où $\mathcal{A}_n$ est la sous-classe des éléments de $\mathcal{A}$ de taille n. Le coefficient de z^n dans la série génératrice $F(z)$ est noté $[z^n]F(z)$.
Remarque: Les séries génératrices usuelles (de dénombrement) sont, au changement de variable $z \to rz$ près, les séries génératrices de probabilités de sources sans mémoire, non biaisées à r symboles.

Il est parfois également utile de disposer de séries génératrices à plusieurs variables (pour marquer la taille et le poids par exemple). Pour cela, on utilise des séries génératrices bivariées définies par

$$F_{\mathcal{A}}(z,u) := \sum_{w\in\mathcal{A}} z^{|w|}u^{c(w)} = \sum_{n\geq 0}\sum_{w\in\mathcal{A}_n} z^n u^{c(w)} = \sum_{n\geq 0}\sum_{k\geq 0} |\mathcal{A}_{n,k}| z^n u^k,$$

où $c(w)$ est une fonction de poids et $\mathcal{A}_{n,k}$ est le sous-ensemble de $\mathcal{A}$ formé des objets de taille n et de poids k.

Ces séries sont les séries de choix lorsqu'on étudie les problèmes sur les mots. Par exemple, l'espérance du nombre d'occurrences $c(w)$ parmi les mots de longueur n produits par une source $\mathcal{S}$ s'obtient en dérivant (par rapport à u) et en extrayant le coefficient de z^n de la série de probabilités

$$F(z,u) = \sum_{w\in\Sigma^\star} p_w u^{c(w)} z^{|w|}.$$

Un dictionnaire pour les sources sans mémoire

Lorsque la source est sans mémoire, la probabilité p_w vérifie la propriété multiplicative suivante

$$\text{si } w = v\cdot v',\ p_w = p_v p_{v'}.$$

Cette propriété se transfère aux séries de probabilités sur les collections, ce qui fournit le dictionnaire suivant :

Collection	Série génératrice
$\mathcal{A}+\mathcal{B}$	$A(z)+B(z)$
$\mathcal{A}\times\mathcal{B}$	$A(z)B(z)$
séquence$(\mathcal{A}) = \mathcal{A}^\star$	$\dfrac{1}{1-A(z)} = \displaystyle\sum_{i\geq 0}(A(z))^i$

Par exemple, les séries génératrices associées à l'ensemble de mots $\Sigma^\star$ où Σ est l'alphabet sont $F(z) = \frac{1}{1-z}$ et $\widetilde{F}(z) = e^z$.

2.2.2 Opérateurs générateurs de mots

Lorsque les symboles du mot ne sont plus indépendants, la propriété de décomposition de la probabilité d'un mot, à la base du dictionnaire algébrique sur les séries génératrices n'est plus vraie. Nous voyons ici comment le cadre exposé précédemment peut tout de même être généralisé grâce à l'utilisation d'opérateurs fonctionnels. Des détails peuvent être trouvés dans ces publications [23, 24, 22, 25].

Dans le cas des sources dynamiques (les chaînes de Markov sont des exemples de sources dynamiques), le problème central consiste à étudier comment évolue la densité de probabilité (celle qui va servir à déterminer quelle sera la prochaine lettre émise). Cette évolution est encodée dans un opérateur fonctionnel appelé opérateur de transfert. C'est une généralisation de la matrice de transition pour les chaînes de Markov.

Dans ce paragraphe, nous définissons différents opérateurs fonctionnels fortement liés aux sources dynamiques puisqu'ils permettent de décrire l'évolution du système dynamique sous-jacent. Ces opérateurs sont dans un sens générateurs puisqu'ils permettent d'obtenir des expressions alternatives pour certains objets qui apparaissent lors de l'étude des différentes séries génératrices pour de sources dynamiques probabilistes.

Transformateur de densité

La forme de base de ces opérateurs est le transformateur de densité. Il a été abondamment étudié. Son but est de décrire l'évolution de la densité de probabilité au cours du temps.

Définition L'opérateur défini par

$$\mathbf{G}[f](x) := \sum_{m\in\Sigma} |h'_m(x)| f\circ h_m(x) \mathbb{1}_{\mathcal{J}_m}(x)$$

est un transformateur de densité. En effet, en considérant l'exemple de la figure 2.6, la densité f_1 en un point y après l'émission du premier symbole par la source dynamique s'exprime à l'aide de la densité initiale f_0 aux points $x_a := h_a(y)$, $x_b := h_b(y)$ et $x_c := h_c(y)$, où h_m désigne l'inverse local de la fonction de décalage T sur $\mathcal{I}_a$. Ainsi, la relation suivante est vérifiée

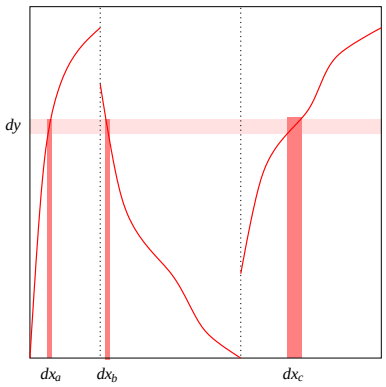
$$|dy| \times f_1(y) := |dx_a| f_0(x_a) + |dx_b| f_0(x_b) + |dx_c| f_0(x_c).$$

En remarquant que $dx_m/dy = h'_m(y)$, on montre que $f_1 = \mathbf{G}[f_0]$. En d'autres termes, l'opérateur $\mathbf{G}$ transforme la densité initiale après une étape du processus. De la même manière, on montre que la densité f_n obtenue après n étapes du processus est $\mathbf{G}^n[f_0]$.

Densité stationnaire La densité stationnaire φ de la source est la limite, si elle existe, de la suite $(\mathbf{G}^n[f])_{n>0}$, où f est une densité analytique quelconque et en particulier,

$$\varphi = \lim_{n\to\infty} \mathbf{G}^n[1].$$

FIGURE 2.6 – Calcul de la densité après l'action de la source.



Cette densité stationnaire est également un point fixe de l'opérateur $\mathbf{G}$, (*i.e.*, $\mathbf{G}[\varphi] = \varphi$). Ceci revient également à dire que 1 est valeur propre de l'opérateur $\mathbf{G}$ associée au vecteur propre φ.

Cette densité stationnaire exprime les probabilités limites d'émission des symboles. Ainsi, elle joue un grand rôle dans les études asymptotiques que nous menons par la suite. L'étude de l'existence et de l'unicité de cette densité invariante constitue un problème central dans l'étude des systèmes dynamiques.

Opérateurs générateurs de mots

Ces opérateurs sont des opérateurs générateurs de probabilités. Ils sont basés sur une troncature de l'opérateur transformateur de densité ou de ses puissances. En effet, l'opérateur $\mathbf{G}$ peut s'écrire sous la forme

$$\mathbf{G} = \sum_{m \in \Sigma} \mathbf{G}_{[m]},$$

où chaque opérateur $\mathbf{G}_{[m]}$ associé au symbole m est défini pour toute fonction positive f et tout réel x de $\mathcal{I}$ par $\mathbf{G}_{[m]}[f](x) := |h'_m(x)| f \circ h_m(x)$ est appelé opérateur de composition.

L'opérateur de composition $\mathbf{G}_{[m]}$ génère la probabilité p_m qu'un mot infini débute par le symbole m. En effet, la définition (2.2) de la probabilité du préfixe m satisfait

$$p_m := \int_{\mathcal{I}_m} f(t)dt = \int_{h_m(\mathcal{J}_m)} f(t)dt = \int_{\mathcal{J}_m} |h'_m(t)| f \circ h_m(t)dt = \int_0^1 \mathbf{G}_{[m]}[f](t)dt,$$

où f est la densité initiale.

Cette remarque s'applique plus généralement à tout préfixe $w \in \Sigma^\star$. L'opérateur générateur de probabilité $\mathbf{G}_{[w]}$ est alors défini pour toute fonction positive f et tout entier x de $\mathcal{I}$ par

$$\mathbf{G}_{[w]}[f](x) := |h'_w(x)| f \circ h_w(x) \mathbb{1}_{\mathcal{J}_w}.$$

Il génère la probabilité p_w dans le sens où

$$p_w = \int_0^1 \mathbf{G}_{[w]}[f](t)dt,$$

où f est la densité initiale de la source.

Cet opérateur est bien évidemment linéaire ce qui permet de le définir également pour des ensembles de mots.

Opérateur générateur d'ensemble et de collection

L'opérateur générateur de mots joue, dans le cas des sources dynamiques, le même rôle que la probabilité p_w elle-même. Ainsi, on peut définir l'analogue des séries génératrices de probabilités associées à un ensemble ou à une collection.

L'*opérateur générateur d'ensemble ou de collection* $\mathbf{A}(z)$ associé à l'ensemble (ou à la collection) $\mathcal{A}$ est défini par

$$\mathbf{A}(z) := \sum_{w\in\mathcal{A}} \mathbf{G}_{[w]} z^{|w|}.$$

Il est lié à la série génératrice de probabilités $A(z)$ par la relation

$$A(z) = \int_0^1 \mathbf{A}(z)[f](t)dt, \tag{2.3}$$

où f est la densité initiale de la source.

Propriété de composition

L'opérateur possède aussi une propriété de composition très importante qui est en quelque sorte l'analogue de la propriété de composition des probabilités $p_{u\cdot v} = p_u p_v$ des sources sans mémoires qui n'est plus vérifiée dans le cas général.

La dérivée vérifie une propriété de composition

$$(f\circ g)'(t) = f'\circ g(t) g'(t),$$

qui, appliquée à l'opérateur générateur de probabilité, se traduit par :

$$\forall u, v \in \Sigma^\star, \mathbf{G}_{[u\cdot v]} = \mathbf{G}_{[v]} \circ \mathbf{G}_{[u]}.$$

On applique simplement la propriété à la fonction $h_{u\cdot v} = h_u \circ h_v$ relative au préfixe $u\cdot v$.

Tout d'abord, cette propriété permet d'affirmer que pour un préfixe w de longueur n, l'opérateur $\mathbf{G}_{[w]}$ est une troncature de la puissance $n^{ème}$ $\mathbf{G}^n$ du transformateur de densité dans le sens où

$$\mathbf{G}^n = \sum_{|w|=n} \mathbf{G}_{[w]}.$$

On peut également fournir le "dictionnaire" suivant :

Ensemble (ou collection)	Opérateur générateur
$\mathcal{A}$	$\mathbf{A}(z) := \sum_{w\in\mathcal{A}} \mathbf{G}_{[w]} z^{\|w\|}$
$\mathcal{A}+\mathcal{B}$	$\mathbf{A}(z)+\mathbf{B}(z)$
$\mathcal{A}\times\mathcal{B}$	$\mathbf{B}(z)\circ\mathbf{A}(z)$
$\mathcal{A}^\star$	$(I-\mathbf{A}(z))^{-1} := \sum_{n>0}\mathbf{A}(z)^n$
Σ	$z\mathbf{G} := z\sum_{m\in\Sigma}\mathbf{G}_{[m]}$
Σ^n	$z^n\sum_{\|w\|=n}\mathbf{G}_{[w]} = z^n\mathbf{G}^n$
$\Sigma^\star$	$(I-z\mathbf{G})^{-1}$

Pour de nombreux problèmes sur les mots, la série génératrice de l'ensemble $\Sigma^\star$ joue un rôle primordial. Dans le cas des sources sans mémoires, cette série possède un pôle simple en $z=1$. Dans le cas des sources dites "bonnes", cette propriété reste vraie et l'opérateur générateur de l'ensemble $\Sigma^\star$ possède lui aussi un pôle simple un $z=1$.

2.2.3 Théorèmes de transfert

Les séries génératrices obtenues précédemment ont été définies d'un point de vue formel. Lorsqu'elles sont considérées comme des fonctions de la variable complexe z, ces fonctions possèdent (éventuellement) des singularités qui dictent le comportement asymptotique des coefficients de la série. Les théorèmes de transfert sont utilisés pour obtenir ce comportement qui dépend ainsi de la position de la singularité et de sa nature. Dans ce mémoire, nous utilisons principalement deux théorèmes de transfert. Que nous nous contentons d'énoncer. Ces théorèmes classiques se retrouvent dans [19].

Extraction de coefficients

En principe, on peut toujours obtenir les coefficient d'une série génératrice en utilisant par exemple un développement de Taylor mais ce procédé peut vite devenir trop complexe pour être utilisé. Le plus souvent, la série génératrice peut-être décomposée en fonctions plus basiques dont le développement en séries entières est parfaitement connu.

Pour les problèmes sur les mots, les séries obtenues se décomposent sous la forme

$$O(z) = \left(\frac{1}{1-z}\right)^{b+1} z^m P(z),$$

où $P(z)$ est une fonction analytique en $z=1$. Cette fonction possède donc un pôle d'ordre $b+1$ en $z=1$. le coefficient de z^n est alors obtenu en utilisant la formule du binôme de Newton

$$[z^n]O(z) = \binom{n-m+b}{b} P(1) = P(1)\frac{(n-m)^b}{b!}\left[1+\frac{b(b+1)}{2(n-m)}O(\frac{1}{n^2})\right]. \quad (2.4)$$

Théorème de Hwang

Il est souvent utile de montrer des convergences en loi. Le théorème des quasi-puissances de Hwang [26] permet de prouver des convergences vers des lois gaus-

siennes en considérant certaines propriétés asymptotiques simples des séries génératrices des moments de la variable aléatoire.

Commençons par quelques définitions.

Définition 2 *Considérons un coût R défini sur un ensemble $\mathcal{R}$ et notons R_n sa restriction au sous-ensemble $\mathcal{R}_n$ des éléments de $\mathcal{R}$ de taille n. Le coût R suit asymptotiquement une loi gaussienne quand $n \to +\infty$ s'il existe trois séquences a_n, b_n, r_n, avec $r_n \to 0$, telles que*

$$\Pr\left[(u,v) \in \mathcal{R}_n \;\middle|\; \frac{R_n(u,v) - a_n}{\sqrt{b_n}} \leq y\right] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{y} e^{-t^2/2}\, dt + O(r[R_n])\,.$$

La séquence r_n définit la vitesse de convergence, notée également $r[R_n]$. La moyenne $\mathrm{E}[R_n]$ et la variance $\mathbf{Var}(R_n)$ vérifient

$$\mathrm{E}[R_n] \sim a_n, \quad \textit{et} \quad \mathbf{Var}(R_n) \sim b_n.$$

Le triplet $(\mathrm{E}[R_n], \mathbf{Var}(R_n), r[R_n])$ est appelé triplet caractéristique de la loi gaussienne de R.

Un théorème communément utilisé pour montrer de telles propriétés de convergence est le théorème suivant dû à Hwang.

Théorème 1 [Hwang] *Soit Z_k une suite de variables aléatoires, dont les séries génératrices des moments $M_k(s)$ admettent l'expression asymptotique*

$$M_k(s) := E[\exp(sZ_k)] = \exp(kU(s) + V(s))(1 + O(1/W_k)), W_k \to \infty,$$

le terme d'erreur étant uniforme sur un disque complexe fermé $|s| \leq s_0$, $s_0 > 0$. Si $U(s)$ et $V(s)$ sont analytiques pour $|s| \leq s_0$ et $U(s)$ satisfait la condition $U''(0) \neq 0$. Alors, la distribution de Z_k est asymptotiquement gaussienne

$$\textit{Pr}\left[\frac{Z_k - kU'(0)}{\sqrt{kU''(0)}} < t\right] = \Phi(t) + O(1/S_k), \textit{ où } \Phi(t) := \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{t} e^{-w^2/2} dw$$

uniformément pour tout x dans $\mathbb{R}$, quand k tend vers ∞, avec $S_k = \min(\sqrt{k}, W_k)$.

3

Problèmes sur les mots et les ensembles de mots : bio-informatique des séquences et génomique

Ce chapitre regroupe plusieurs résultats sur des problématiques liées aux mots eux-mêmes ou aux ensembles de mots. Le but ici est d'illustrer l'importance des résultats d'analyse en moyenne d'algorithmes pour répondre à des questions très appliquées en bioinformatique. Le premier problème exposé concerne la comparaison de séquences, en fournissant un indice de similarité. Plusieurs méthodes de pondération de cette similarité sont passées en revue et des résultats statistiques fins sont obtenus. Ce premier problème permet de dresser un large panorama des méthodes utilisées dans l'étude des séquences. Il est exposé de manière détaillée. Le second problème concerne l'obtention de critères statistiques pour la recherche de motifs. Le but ici est de répondre à des questions du type, si j'observe x occurrences d'un motif donné, est-ce significatif ? Les réponses apportées dépendent de deux choses principalement, le modèle aléatoire qui est censé être fidèle à la réalité et le type de motif recherché. Enfin, nous présentons un dernier problème qui concerne l'étude de structures d'index et qui trouve un intérêt important lorsqu'il s'agit de comparer les efficacités des algorithmes qui manipulent ces structures.

3.1 Statistiques des similarités entre mots

En bioinformatique, de nombreux algorithmes ont été développés pour résoudre le problème de la découverte de motifs dans un ensemble de séquences biologiques. Les motifs découverts étant généralement liés à des propriétés biologiques importantes comme la présence d'un site de fixation de facteur de transcription par exemple.

L'idée principale de la plupart de ces algorithmes est qu'un motif important

(au sens biologique) est un motif sur-représenté dans un ensemble de séquences biologiques, selon un critère de similarité entre les séquences biologiques. Il faut noter qu'à cause de transformations des séquences biologiques au cours de l'évolution (eg. mutations), les motifs recherchés ne sont jamais parfaitement conservés entre les séquences. Ainsi, à une certaine étape de leur déroulement, les algorithmes doivent décider si deux motifs, extraits de deux séquences différentes, sont similaires ou non. De plus, les algorithmes nécessitent également la plupart du temps des résultats plus précis permettant de décider si deux motifs sont plus similaires que d'autres. De nombreux algorithmes sont confrontés à cette question qui est alors résolue de manière plus ou moins heuristique. Dans cette partie, tirée de l'article [6], je me suis appliqué à fournir une réponse rigoureuse permettant d'avoir une mesure de similarité normalisée. Ceci se traduit par un calcul de moyenne, écart-type et distribution de la similarité entre deux motifs dont peuvent être dérivés des formules calculables rapidement pour des critères statistiques comme des Z-scores et des p-valeurs.

3.1.1 Les similarités entre séquences

On considère le problème suivant : soient deux séquences $w^{[1]}$ et $w^{[2]}$ de même longueur n. On note w le mot de $\{0,1\}$ de longueur n tel que $w_i = 1$ si, et seulement si, $w_i^{[1]} = w_i^{[2]}$. Le mot w est donc de la forme

$$w = 0^{k_0}1^{\ell_1}0^{k_1}\cdots 1^{\ell_t}0^{k_t}1^{\ell_{t+1}},$$

où $t \geq 0$, $\forall i \in \{1,\ldots,t\}, k_i, \ell_i > 0$ et $k_0, \ell_{t+1} \geq 0$. On appelle *suite des matches* la suite

$$matches(w^{[1]}, w^{[2]}) \equiv matches(w) := (\ell_1, \ldots, \ell_{t+1})$$

et *suite des mismatches* la suite

$$mismatches(w^{[1]}, w^{[2]}) \equiv mismatches(w) := (k_0, \ldots, k_t).$$

On dispose en outre de deux fonctions $f^{=}$ et $f^{\neq}$ qui sont toutes deux des fonctions strictement monotones sur $[0, +\infty[$ et telles que $\lim_{k\to\infty} f(k)/2^k = 0$ (ou qui satisfont la condition plus forte $f(k) = o(\sqrt{\exp(x)})$). On supposera en outre que $f(0) = 0$. Ces deux fonctions seront appelées *fonctions composantes du score*.

Le *score* des deux séquences (noté $S(w^{[1]}, w^{[2]})$) est égal à la somme des valeurs de la fonction $f^{=}$ (resp. $f^{\neq}$) appliquée à tous les matches (resp. tous les mismatches),

$$S(w^{[1]}, w^{[2]}) \equiv S(w) = \sum_{\ell \in matches(w)} f^{=}(\ell) + \sum_{k \in mismatches(w)} f^{\neq}(k). \qquad (3.1)$$

Cette fonction de score est celle utilisée par le logiciel d'extraction de motifs biologiques STARS [6].

La fonction de score $S(w)$ n'est pas additive, toutefois, elle est additive "par blocs" (par définition) dans le sens suivant : $S(u \cdot v) = S(u) + S(v)$ à condition que la dernière lettre de u soit différente de la première lettre de v. Cette notion faible d'additivité sera suffisante.

Lorsque les deux séquences $w^{[1]}$ et $w^{[2]}$ sont des séquences aléatoires, le score noté $S(w^{[1]}, w^{[2]})$ lui-même est une variable aléatoire à valeur dans $\mathbb{R}$. On étudie ici la moyenne, la variance et la distribution limite de cette variable aléatoire lorsque les séquences sont produites chacune par une source sans mémoire de probabilités respectives $\{p_\alpha^{[1]}\}_{\alpha\in\Sigma}$ et $\{p_\alpha^{[2]}\}_{\alpha\in\Sigma}$.

Plus précisément, on va étudier la variable aléatoire $S(w)$ où w est un mot aléatoire produit par une source sans mémoire de probabilités $\{p_0 := 1-p, p_1 = p\}$. On se ramène à l'étude du score entre deux mots en utilisant la valeur $p = \sum_{\alpha\in\Sigma} p_\alpha^{[1]} p_\alpha^{[2]}$.

3.1.2 Principe général de la méthode

Les principales caractéristiques de la méthode sont exposées maintenant.

Rappels. Soit $\mathcal{L}$ un ensemble de mots. La *série génératrice* (probabilisée) simple associée à l'ensemble $\mathcal{L}$ est la série (entière) formelle $L(z)$ suivante :

$$L(z) := \sum_{w\in\mathcal{L}} p_w z^{|w|} = \sum_{n\geq 0} z^n \sum_{w\in\mathcal{L},|w|=n} p_w,$$

où $|w|$ est la longueur du mot w et p_w en est sa probabilité. La notation $[z^n]L(z)$ va désigner le cœfficient de z^n dans la série formelle $L(z)$.

On se donne une fonction de coût $S(w)$ d'un mot w. La série génératrice double (probabilisée) du coût associée à l'ensemble $\mathcal{L}$ est la série formelle $L(z,u)$ suivante :

$$L(z,u) := \sum_{w\in\mathcal{L}} p_w u^{S(w)} z^{|w|} = \sum_{n\geq 0} z^n \sum_{w\in\mathcal{L},|w|=n} p_w u^{S(w)}.$$

Moyenne et variance du coût. Dans la suite, on va s'intéresser à la moyenne et à la variance du coût $S(w)$, lorsque w est un mot aléatoire de l'ensemble $\Sigma^\star = \{0,1\}^\star$ de longueur n (la variable aléatoire correspondante sera notée S_n). Ces deux quantités s'obtiennent facilement à partir de la série génératrice double. En effet, on a :

$$\mathrm{E}\left[S_n\right] := \sum_{|w|=n} p_w S(w) = [z^n]\ \frac{\partial}{\partial u} L(z,u)\bigg|_{u=1}, \text{ et}$$

$$\mathbf{Var}(S_n) = \mathrm{E}\left[S_n^2\right] - \left(\mathrm{E}\left[S_n\right]\right)^2, \text{ avec}$$

$$\mathrm{E}\left[S_n^2\right] := \sum_{|w|=n} p_w S(w)^2 = [z^n]\left(\frac{\partial^2}{\partial u^2} L(z,u)\bigg|_{u=1} + \frac{\partial}{\partial u} L(z,u)\bigg|_{u=1}\right).$$

Tout le problème revient donc à trouver une expression de $L(z,u)$ et de ses dérivées de laquelle on pourra facilement extraire le coefficient de z^n via les théorèmes de transfert de la section 2.2.3.

Un exemple simple. On se place dans le contexte simple où la source est sans mémoire et produit un 1 avec la probabilité p. Les fonctions de score considérées sont les fonctions linéaires $f^{=}(k) = \alpha k$ et $f^{\neq}(k) = -\beta k$ La fonction de score $S(w)$ correspondante est donc additive (chaque symbole à une contribution fixée au score égale à α si c'est un 1 et $-\beta$ sinon).

Dans ce cas, le langage que l'on considère est $\Sigma^\star = (\{0\} \cup \{1\})^\star$. La série génératrice associée s'écrit donc

$$L(z,u) = \frac{1}{1 - z(p_0 u^{f^{\neq}(1)} + p_1 u^{f^{=}(1)})}.$$

Si on s'intéresse à la moyenne,

$$\frac{\partial}{\partial u} L(z,u)\bigg|_{u=1} = \frac{z(\alpha p_1 - \beta p_0)}{(1-z)^2}.$$

Ainsi, en notant que $[z^n](1-z)^{-2} = n+1$, on obtient la moyenne :

$$\mathrm{E}\,[S_n] = n(\alpha p_1 - \beta p_0).$$

Si on s'intéresse à la variance, on a :

$$\left(\frac{\partial^2}{\partial u^2} + \frac{\partial}{\partial u}\right) L(z,u)\bigg|_{u=1} = \frac{z c_2}{(1-z)^2} + \frac{2 z c_1^2}{(1-z)^3},$$

où $c_1 = \alpha p_1 - \beta p_0$ et $c_2 = \alpha^2 p_1 + \beta^2 p_0$.

En remarquant que $[z^n](1-z)^{-3} = (n+1)(n+2)/2$, on obtient :

$$\mathbf{Var}(S_n) = n c_2 + n(n+1)c_1^2 - n^2 c_1^2 = n(c_1^2 + c_2).$$

Pour prouver que la distribution est asymptotiquement gaussienne, on peut appliquer le théorème de Hwang.

Dans la suite, notre but sera de prouver que pour n'importe quelle autre fonction de score $S(w)$ (telle qu'elle a été définie en (3.1)), l'espérance et la variance sont également d'ordre n.

La bonne décomposition. Comme dans le cas général, la fonction de score n'est pas additive, on ne peut pas utiliser la décomposition précédente. Cependant, on peut tout de même remarquer que les fonctions de score sont additives "par bloc", (*i.e.*, $S(u \cdot v) = S(u) + S(v)$ à condition que la dernière lettre de u soit différente de la première lettre de v). Ainsi, toute décomposition qui respecte les blocs pourra être utilisée.

Notons tout d'abord que tout mot de $\{0,1\}^\star$ se décompose en des suites consécutives de 0 et de 1. Autrement dit, on a

$$\{0,1\}^\star = 0^\star (1^+ 0^+)^\star 1^\star.$$

Si on note respectivement $S_0(z,u) := \sum_{k>0} p_0^k u^{f^{\neq}(k)} z^k$ et $S_1(z,u) := \sum_{k>0} p_1^k u^{f^{=}(k)} z^k$ les séries génératrices associées aux ensembles 0^+ et 1^+, la série génératrice $L(z,u)$ de l'ensemble $\{0,1\}^\star$ s'écrit sous la forme :

$$L(z,u) = (1 + S_0(z,u)) \cdot \frac{1}{1 - S_1(z,u) S_0(z,u)} \cdot (1 + S_1(z,u)).$$

L'étude de la moyenne et de la variance se base donc sur ces séries génératrices dont on peut montrer que

$$S_0(z,1) = \frac{p_0 z}{1-p_0 z}, \qquad S_1(z,1) = \frac{p_1 z}{1-p_1 z},$$
$$\frac{\partial}{\partial u} S_0(z,u)\bigg|_{u=1} = \sum_{k>0} p_0^k f^{\neq}(k) z^k, \qquad \text{et}$$
$$\left(\frac{\partial^2}{\partial u^2} + \frac{\partial}{\partial u}\right) S_0(z,u)\bigg|_{u=1} = \sum_{k>0} p_0^k (f^{\neq}(k))^2 z^k$$

sont des séries analytiques sur $\mathbb{R}$ à condition que $f^{\neq}$ ne soit pas trop grande (en ordre, typiquement $f^{\neq}(x) = o(\sqrt{\exp x})$). Cette dernière propriété est également vérifiée par les dérivées en u de $S_1(z,u)$.

3.1.3 Etude du score moyen

Cas général. On est amené ici à calculer la dérivée première en $u = 1$ de $L(z,u)$. Afin d'alléger les expressions, on adopte les raccourcis de notation suivants :

$$S_0 := S_0(z,u), \qquad S_1 := S_1(z,u), \qquad S_0' := \frac{\partial}{\partial u} S_0(z,u), \qquad S_1' := \frac{\partial}{\partial u} S_1(z,u).$$

La dérivée partielle par rapport à u s'écrit donc

$$\frac{\partial}{\partial u} L(z,u) = \frac{[S_0'(1+S_1) + S_1'(1+S_0)](1-S_0S_1) + (1+S_0)(1+S_1)(S_1'S_0 + S_0'S_1)}{(1-S_0S_1)^2},$$

ce qui se simplifie et on obtient :

$$\frac{\partial}{\partial u} L(z,u) = \frac{S_0'(1+S_1)^2 + S_1'(1+S_0)^2}{(1-S_0S_1)^2}. \tag{3.2}$$

Lorsque cette dérivée est évaluée en $u = 1$, toutes les quantités qui font intervenir S_0 ou S_1 ont une expression particulièrement simple :

$$(1+S_0) = \frac{1}{1-zp_0}, \qquad (1+S_1) = \frac{1}{1-zp_1}, \qquad \frac{1}{1-S_1S_0} = \frac{(1-zp_0)(1-zp_1)}{1-z}. \tag{3.3}$$

En outre, on peut remarquer que les fonctions (S_0'/z) et (S_1'/z) sont des fonctions analytiques sur $\mathbb{R}$ qui ne s'annulent ni en 0, ni en 1. La dérivée s'écrit donc :

$$\frac{\partial}{\partial u} L(z,u)\bigg|_{u=1} = \left(\frac{1}{1-z}\right)^2 z^1 [S_0'(1-zp_0)^2 + S_1'(1-zp_1)^2].$$

Le résultat d'extraction de coefficient de la section 2.2.3 s'applique clairement et on obtient la moyenne suivante :

$$\mathrm{E}\,[S_n] = n(s_0' p_1^2 + s_1' p_0^2), \tag{3.4}$$

où $s_0' := S_0'|_{z=1,u=1}$ et $s_1' := S_1'|_{z=1,u=1}$ sont deux quantités qui font intervenir les fonctions $f^{\neq}$ et $f^{=}$ de la manière suivante :

$$s_0' = \sum_{k>0} f^{\neq}(k) p_0^k, \qquad s_1' = \sum_{k>0} f^{=}(k) p_1^k.$$

Application à quelques fonctions classiques. Commençons par montrer que ce résultat pour la moyenne est conforme à celui obtenu dans la première partie lorsqu'on s'intéresse à deux fonctions linéaires simples pour $f^{\neq}$ et $f^{=}$ ($f^{\neq}(k) = -\beta k$ et $f^{=}(k) = \alpha k$). Si on note $D(z) := 1/(1-z)$, on a

$$s'(k \mapsto \alpha \cdot k, p) = \sum_{k>0} \alpha k p^k = \alpha \ (zD'(z))|_{z=p} = \alpha \frac{p}{(1-p)^2}.$$

On retrouve bien la formule attendue, c'est à dire $s'_0 p_1^2 = -\beta p_0$ et $s'_0 p_1^2 = \alpha p_1$.

On s'intéresse maintenant aux fonctions quadratiques. Avec la même définition que précédemment pour $D(z)$, on remarque que

$$s'(k \mapsto k^2, p) = \sum_{k>0} k^2 p^k = \ (z^2 D''(z) + zD'(z))\big|_{z=p} = \frac{p(p+1)}{(1-p)^3}.$$

Le même type de calcul est également valable pour les fonctions cubiques et on a :

$$s'(k \mapsto k^2, p) = \sum_{k>0} k^2 p^k = \ (zD'(z) + 3z^2 D''(z) + z^3 D^{(3)}(z))\big|_{z=p} = \frac{p(1+4p+p^2)}{(1-p)^4}.$$

Clairement, le raisonnement s'étend à n'importe quelle puissance entière et moyennant certains efforts de calcul, on peut obtenir une formule close pour $s'(k \mapsto k^\ell, p)$, quel que soit ℓ entier.

Il y a une propriété (évidente) de linéarité pour les fonctions composantes de score dans le sens suivant :

$$s'(k \mapsto \alpha f_1(k) + \beta f_2(k), p) = \alpha \cdot s'(k \mapsto f_1(k), p) + \beta \cdot s'(k \mapsto f_2(k), p).$$

Le cas des fonctions constantes est un tout petit peu différent puisqu'en toute rigueur, elles ne vérifient pas les conditions requises (elles ne sont pas strictement croissantes). Pourtant, les deux fonctions constantes $k \mapsto 0$ et $k \mapsto 1$ sont très intéressantes puisque la première permet de ne pas prendre en compte les matches (ou les mismatches) dans le calcul du score tandis que la deuxième permet de compter le nombre de suites de matches (ou de mismatches). On reprenant les lignes du calcul, on remarque qu'il s'adapte aussi à ces fonctions et on montre facilement que le résultat reste valide avec une constante

$$s'(k \mapsto \alpha, p) = \sum_{k>0} \alpha p^k = \alpha \ D(z)|_{z=p} = \alpha \frac{p}{1-p}.$$

Pour les fonctions plus "exotiques" (comme $k \mapsto \sqrt{k}$ ou $k \mapsto \ln(1+k)$, il est difficile d'obtenir une formule close. Cependant, le terme général $f(k)p^k$ de la série tends très rapidement vers 0 et il suffit de calculer très peu de termes pour obtenir une bonne précision dans le calcul. Cette précision peut être définie quantitativement par le critère de Leibniz.

3.1.4 Etude de la variance du score

Dans cette partie, aucun outil supplémentaire n'est nécessaire. Le calcul est juste légèrement plus intriqué que dans le cas de la moyenne.

Cas général. On s'intéresse ici à la dérivée seconde (par rapport à u) de $L(z,u)$ en $u=1$. Partons de l'expression de la dérivée première obtenue en (3.2). Comme dans la partie précédente, les quantités S_0, S_1, S_0', S_1', S_0'' et S_1'' désignent les fonctions à deux variables S_0 ou S_1 et leurs dérivées par rapport à u.

On obtient alors l'expression

$$\begin{aligned}\left(\frac{\partial^2}{\partial u^2}+\frac{\partial}{\partial u}\right)L(z,u) = \quad & \frac{1}{(1-S_0S_1)^2}\times \quad [(S_0''+S_0')(1+S_1)^2+(S_1''+S_1')(1+S_0)^2 \\ & \qquad +2S_0'S_1'(S_0+S_1+3S_0S_1) \\ & \qquad +2(S_0')^2S_1(1+S_1)+2(S_1')^2S_0(1+S_0)] \\ & -\frac{1}{(1-S_0S_1)^3} \quad [2(S_0+1)(S_1+1)(S_0'S_1+S_1'S_0)].\end{aligned}$$

En $u=1$, on traite les lignes unes par unes.

Tout calcul fait, on obtient une variance égale à

$$\begin{aligned}\mathbf{Var}(S_n) &= n[(s_0''+s_0')p_1^2+(s_1''+s_1')p_0^2+(s_0')^2p_1^3(1+5p_0)+(s_1')^2p_0^3(1+5p_1) \\ &+ 2p_1s_0's_1'(4-9p_1+10p_1^2-10p_1^3)+s_1''p_0(1+p_1)+s_0''p_1(1+p_0) \\ &+ 6p_1^2p_0^2((s_0')^2+(s_1')^2)+4s_0's_1'(p_0^2+p_1^2+3p_1^2p_0^2)],\end{aligned}$$

où les constantes s_0'', s_1'', s_0', s_1' se calculent (au moins) rapidement et admettent les expressions suivantes :

$$s_0'' := \sum_{k>0} p_0^k f^{\neq}(k)(f^{\neq}(k)-1), \quad s_1'' := \sum_{k>0} p_1^k f^{=}(k)(f^{=}(k)-1),$$

$$s_0' := \sum_{k>0} p_0^k f^{\neq}(k), \quad s_1' := \sum_{k>0} p_1^k f^{=}(k).$$

Cas particulier. Comme dans le cas de la moyenne, les constantes qui apparaissent dans l'expression de la variance admettent des formules closes lorsque les fonctions composantes de score sont simples. On va donc chercher une formule close pour l'expression :

$$s''(k\mapsto f(k),p) := \sum_{k>0} p^k f(k)(f(k)-1)$$

pour des fonctions f données. Pour trouver de telles expressions, on considèrera la fonction $D(z) := 1/(1-z)$ et ses dérivées.

Dans le cas des fonctions affines, on a

$$\begin{aligned}s''(k\mapsto \alpha k,p) &= \alpha^2\,(zD'(z)+z^2D''(z))\big|_{z=p} - \alpha\,(zD'(z))|_{z=p} \\ &= \frac{\alpha p((\alpha+1)p+\alpha-1)}{(1-p)^3}.\end{aligned}$$

Dans le cas des fonctions carrés, si on note Δ l'opérateur $z\frac{d}{dz}$, on a

$$s''(k\mapsto k^2,p) = \Delta^4\,D(z)|_{z=p} - \Delta^2 D(z)\big|_{z=p} = \frac{p(11+12p+p^2)}{(1-p)^5}.$$

Plus généralement, pour toute fonction monomiale $k \mapsto k^m$, on a la relation suivante qui se preuve par une simple récurrence :

$$s''(k \mapsto k^2, p) = \Delta^{2m} D(z)\big|_{z=p} - \Delta^m D(z)|_{z=p}\,.$$

Il n'y a pas la même notion de linéarité que dans le cas de la constante s'. Cependant, on peut tout de même montrer que

$$s''(k \mapsto (f{+}g)(k), p) = s''(k \mapsto f(k), p){+}s''(k \mapsto g(k), p){-}2s'(k \mapsto (f{\times}g)(k), p).$$

3.1.5 Distribution limite du score

Dans certains cas particuliers, lorsque les fonctions qui composent le score sont linéaires, le score se comporte comme une variable Binomiale dont la distribution limite est une gaussienne.

J'étends ici ce résultat à n'importe quel couple de fonctions composante de score. La preuve se décompose en plusieurs étapes : introduction d'une nouvelle suite de variables aléatoires dont on peut prouver qu'elle suit la même distribution limite que le score et preuve que cette nouvelle suite de variables aléatoires suit une loi normale (par l'application du théorème central limite).

Le théorème suivant sera donc prouvé.

Théorème 2 *Lorsque la similarité est produite par une source sans mémoire de paramètre p, la variable aléatoire S_n admet le résultat suivant de convergence en loi*

$$\frac{S_n - \frac{nc'}{c}}{\sqrt{\mathbf{Var}(S_n)}} \xrightarrow[n\to\infty]{(d)} \mathcal{N}(0,1),$$

où $c = E_0 + E_1 = \frac{1}{p(1-p)}$, $c' = E'_0 + E'_1$ and $\mathbf{Var}(S_n)$ admettent les expressions obtenues précédemment.

Rappelons ce lemme classique :

Lemme 1 *Soient (X_n) et (Y_n) deux séquences de variables aléatoires dans $\mathbb{R}^d$, si $X_n \xrightarrow[n]{\mathcal{L}} X$ et $||X_n - Y_n|| \xrightarrow[n]{proba} 0$ (pour une norme donnée sur $\mathbb{R}^d$), alors $Y_n \xrightarrow[n]{\mathcal{L}} X$.*

Preuve:[du théorème]

Nous introduisons deux marches aléatoires centrées qui seront utiles pour décomposer les quantités d'intérêt.

$$Z_k = \sum_{i=1}^{k} L_i^{(0)} + L_i^{(1)} - c, \qquad \text{et } Z'_k = \sum_{i=1}^{k} f^{(0)}(L_i^{(0)}) + f^{(1)}(L_i^{(1)}) - c',$$

où $c = E_0 + E_1 = \frac{1}{p(1-p)}$ and $c' = E'_0 + E'_1$.

La variable aléatoire S_n vérifie :

$$S_n - \frac{nc'}{c} = \left(Z'_{\tau_n} - Z'_{n/c}\right) + Z'_{n/c} + \left(\tau_n - \frac{n}{c}\right) c' + R_n.$$

Le théorème 2 est une conséquence de la proposition suivante :

Proposition 1 *Les convergences suivantes sont satisfaites*

(1) $n^{-1/2}R_n \xrightarrow[n]{proba} 0$ *;*

(2) $n^{-1/2}\sqrt{c}\left(Z'_{n/c}\,,\ (\tau_n - \frac{n}{c})c'\right) \xrightarrow[n]{\mathcal{L}} \mathcal{N}(0, M)$*, La distribution gaussienne centrée sur* $\mathbb{R}^2$ *de matrice de covariance*

$$M = \begin{pmatrix} V'_0 + V'_1 & (c'/c)\rho \\ (c'/c)\rho & (c'/c)^2(V_0 + V_1) \end{pmatrix}$$

où $\rho = \frac{c'}{c}\, cov\left(f^{(0)}(L_1^{(0)}) + f^{(1)}(L_1^{(1)}), L_1^{(0)} + L_1^{(1)}\right)$,

(3) $n^{-1/2}(Z'_{\tau_n} - Z'_{n/c}) \xrightarrow[n]{proba} 0$.

En effet, on a

$$n^{-1/2}\left(Z'_{n/c} + (\tau_n - \frac{n}{c})c'\right) \xrightarrow[n]{\mathcal{L}} \frac{1}{\sqrt{c}}\mathcal{N}(0, \Sigma^2),$$

où la variance $\Sigma^2 = \frac{1}{c}((c'/c)^2(V_0+V_1)+(V'_0+V'_1)+2(c'/c)\rho$. Il est aisé de vérifier que cette quantité admet le même terme de premier ordre que $\mathbf{Var}(S_n)$ donné en Section précédente. Le lemme 1 permet de conclure la preuve du théorème. □

Nous prouvons maintenant tous les points de la proposition.

Preuve: (1) Puisque $f^{(0)}$ et $f^{(1)}$ sont d'ordre polynomial, il existe κ tel que $f^{(0)}(m) \leq m^\kappa$ et $f^{(1)}(m) \leq m^\kappa$. On a

$$|R_n| = |f^{(0)}(K_n^{(0)})+f^{(1)}(K_n^{(1)})| \leq |(K_n^{(0)})^\kappa|+|(K_n^{(1)})^\kappa| \leq |\sup_{1\leq n/2}(L_i^{(0)})^\kappa|+\sup_{1\leq n/2}|(L_i^{(1)})^\kappa|,$$

car $K_n^{(0)}$ et $K_n^{(1)}$ sont inclues dans un des n premiers blocs.

Maintenant, nous montrons que la probabilité que le maximum de $n/2$ variables aléatoires géométriques soit supérieur à $\varepsilon\sqrt{n}$ tend vers 0 lorsque $n \to +\infty$ (en fait, ce maximum est d'ordre $\log n$).

(2) Pour commencer, la définition de τ_n implique que

$$\left\{\frac{(\tau_n - n/c)c'}{\sqrt{n}} \leq y\right\} = \left\{Z_{\frac{n}{c}+y\frac{\sqrt{n}}{c'}} \geq -y\sqrt{n}\frac{c}{c'}\right\}.$$

L'inégalité de Bienaymé-Tchebyshev implique $n^{-1/2}(Z_{\frac{n}{c}} - Z_{\frac{n}{c}+y\frac{\sqrt{n}}{c'}}) \xrightarrow[n]{proba.} 0$, et ensuite, le lemme 1 peut-être utilisé pour obtenir que $X_n = (Z'_{\frac{n}{c}}, Z_{\frac{n}{c}})$ et $Y_n = (Z'_{\frac{n}{c}}, Z_{\frac{n}{c}+y\frac{\sqrt{n}}{c'}})$ ont la même distribution limite dans $\mathbb{R}^2$, si elle existe. Maintenant, le vecteur $X_n = (Z'_n, Z_n)$ est clairement la somme de n variables aléatoires indépendante et identiquement distribuées (et centrées) Γ_i avec

$$\Gamma_i = (f^{(0)}(L_i^{(0)}) + f^{(1)}(L_i^{(1)}) - c', L_i^{(0)} + L_i^{(1)} - c).$$

Le résultat est donc une conséquence du théorème central limite appliqué à $X_{n/c}$.

(3) Soit $\varepsilon > 0$. Nous allons établir que $\lim_{n\to\infty}$ Prob $\left\{|Z'_{\tau_n} - Z'_{n/c}| \geq \varepsilon\sqrt{n}\right\} = 0$. Distinguons deux cas selon que la condition $|\tau_n - n/c| \leq n^{2/3}$ est satisfaite ou non. On a

$$\text{Prob}\left\{\frac{|Z'_{\tau_n} - Z'_{n/c}|}{\sqrt{n}} \geq \varepsilon\right\} \leq \text{Prob}\left\{|Z'_{\tau_n} - Z'_{n/c}| \geq \varepsilon\sqrt{n}, |\tau_n - n/c| \leq n^{2/3}\right\} + \text{Prob}\left\{|\tau_n - n/c| > n^{2/3}\right\}.$$

La seconde probabilité tend vers 0 lorsque $n \to \infty$ par (2). La première probabilité que nous notons maintenant a_n vérifie

$$\begin{aligned} a_n &\leq \text{Prob}\left\{\Delta(Z', [n/c - n^{2/3}, n/c + n^{2/3}]) \geq \varepsilon\sqrt{n}\right\} && (3.5)\\ &= \text{Prob}\left\{\Delta(Z', [0, 2n^{2/3}] \geq \varepsilon\sqrt{n}\right\} && (3.6)\\ &\leq \text{Prob}\left\{\max\{2|Z'_k|, k \in [0, 2n^{2/3}]\} \geq \varepsilon\sqrt{n}\right\} && (3.7) \end{aligned}$$

pour tout interval I, $\Delta(Z', I) = \max\{Z'_k, k \in I\} - \min\{Z'_k, k \in I\}$. La formule (3.5) est une conséquence des faits suivants. Premièrement, si $I \subset J$ alors $\Delta(Z', I) \leq \Delta(Z', J)$. Ensuite, $|Z'_{\tau_n} - Z'_{n/c}| \leq \Delta(Z', [\tau_n \wedge n/c, \tau_n \vee n/c])$. Ainsi, $|Z'_{\tau_n} - Z'_{n/c}| \leq \Delta(Z', [n/c - n^{2/3}, n/c + n^{2/3}])$ lorsque $|\tau_n - n/c| \leq n^{2/3}$. L'équation (3.6) vérifie la propriété de Markov pour la marche aléatoire Z', et (3.7) est clair.

L'inégalité de Doob s'applique à la martingale (Z'_k) prouvant que pour tout $q > 1$,

$$\text{E}\left[\max_{0\leq k\leq m} |Z'_k|^q\right] \leq (\frac{q}{q-1})^q \text{E}\left[|Z'_m|^q\right].$$

De plus, par (3.7) et en utilisant l'inégalité de Markov, en prenant $m = 2n^{2/3}$ et $q = 2$,

$$a_n \leq \text{Prob}\left\{\max_{k\in[0,m]} |Z'_k|^2 \geq \varepsilon^2 n/4\right\} \leq C\frac{\text{E}\left[|Z'_m|^2\right]}{\varepsilon n} = \frac{Cm\sigma'^2}{\varepsilon n},$$

pour une constante C. Cette dernière quantité converge vers 0 lorsque $n \to \infty$. □

3.1.6 Conclusion et prespectives

Les résultats présentés ici peuvent être complétés de plusieurs manières. Premièrement, cette étude concerne uniquement les cas de sources aléatoires sans mémoire. Cependant, pour se rapprocher de la réalité biologique, il serait intéressant d'étudier des modèles probabilistes plus riches comme les sources dynamiques. Ces résultats peuvent être étendus à ces sources. En effet, le point clé de l'étude réside dans le fait que les expressions de S_0, S_1 et de leurs dérivées en $u = 1$ sont très simples. Ceci reste vrai lorsque les séries génératrices sont remplacées par des opérateur générateurs. Des résultats, qui mettraient en avant les influences des corrélations de la sources, pourraient être obtenus.

Ensuite, pour calculer le score entre deux mots $w^{[1]}$ et $w^{[2]}$, Seulement deux types d'évènements sont considérés les "matches" et les "mismatches". Il est facile d'imaginer d'autres évènements. Par exemple lorsque l'on cherche à calculer une

similarité sur un ensemble de mots (de taille m), il est en général utile d'ajouter un quorum Q qui conduit à définir les matches et les mismatches différemment. A une position, il y aura un match s'il y a une lettre commune à plus de $Q\%$ des séquences, ce sera un mismatch si plus de $Q\%$ des séquences contiennent des lettres différentes (autrement dit la lettre majoritaire est commune à moins de $(100 - Q\%)$ des séquences. Dans les autres cas plus litigieux, la position ne doit pas intervenir dans le calcul (dans la similarité, elle sera codée par un troisième symbole).

Ici, on s'intéresse donc aux propriétés d'un mot construit sur un alphabet à trois lettres $\{0, 1, x\}$ où les runs sont quantifiés par trois fonctions f_0, f_1 et f_x (la plupart du temps, on souhaite que $f_x \equiv 0$ ne compte pas dans la statistique). Pour étudier ce nouveau problème, il s'agit d'adapter la décomposition de base de $\Sigma^\star$. Cependant, le coeur de la méthode reste valide.

3.2 Statistiques de motifs complexes

Le problème de recherche de motifs particuliers dans un texte est particulièrement important en théorie de l'information. Ainsi, il est intéressant d'étudier précisément le nombre d'occurrences d'un motif donné dans un texte "typique". Ici, "typique" signifie principalement que le texte est aléatoirement produit par une source probabiliste qui reproduit le plus fidèlement possible la complexité des séquences réelles qui sont étudiées. Une telle étude du nombre d'occurrences permet d'obtenir des résultats sur la complexité moyenne des algorithmes de recherche de motifs. Cela fournit également des heuristiques statistiques précise (comme des Z-scores ou des p-valeurs) qui aident à interpréter les résultats de ces algorithmes voire même d'en améliorer la complexité. Il est aussi intéressant de considérer le nombre de positions d'occurrence (i.e., le nombre de positions dans le texte où une occurrence du motif se termine).

Ces deux paramètres, notés dans la suite Ω pour le nombre d'occurrences et C pour le nombre de positions d'occurrences peuvent être significativement différents, notamment car le nombre de positions d'occurrence est nécessairement borné par la longueur du texte cible ce qui n'est pas vrai pour le nombre d'occurrences (il peut exister plusieurs occurrences qui se terminent à la même position, il se peut même, dans certains cas d'expressions régulières étudiées par exemple dans [8], que deux occurrences différentes dans le texte commencent et terminent aux mêmes positions).

Dans le paragraphe suivant, nous présentons quelques grandes familles de motifs sur les séquences. Des résultats asymptotiques ont été obtenus pour chacune de ces familles, dans des modèles probabilistes très divers également. Dans la suite, nous nous focalisons sur l'étude des positions d'occurrences d'expressions régulières dans un modèle de sources dynamiques.

3.2.1 Les différents motifs

Le problème de base lorsque l'on considère la recherche de motif consiste à chercher un mot.

Recherche de mot. Ici, le mot w est recherché dans le texte cible T comme une suite consécutive de symboles (s'il s'y retrouve, on dira que c'est un *facteur* du texte).

Motif	*Texte*	*occ*	*pos occ*
abr	abracadadabra	2	2
ada	abracadadabra	2	2

Lors de l'étude, les autocorrélations du motif (qui impliquent qu'il y a deux occurrences de ada dans adada jouent un rôle important.

Recherche d'ensemble (fini) de mots. L'extension naturelle à ce problème consiste à rechercher un ensemble fini W de tels mots en tant que facteurs du texte. La recherche de motifs approchée ou encore la recherche de motifs de type matrice poids positions (très populaire en bioinformatique) tombent dans ce cadre.

Motif	*Texte*	*occ*	*pos occ*
{abr,ada}	abracadadabra	4	4
{da,ada}	abracadadabra	4	2

La notion d'autocorrélation se généralise ici à une matrice de corrélations. De plus, dans le cas d'ensemble de motifs, il devient possible qu'une position d'occurrence ne corresponde plus à une unique occurrence de motifs, ce qui rend l'étude du problème légèrement plus compliquée.

Recherche de séquences. Ici, on recherche toujours un mot dans un texte cible mais les symboles du motif n'ont plus à être consécutifs (le motif est cherché en tant que sous séquence du texte). Ce problème, parfois appelé recherche de motif caché, peut-être étendu en ajoutant des bornes sur certaines distances entre symboles successifs et même en considérant des séquences de langages à la place de séquences de symboles (ce dernier problème est connu sur le nom de recherche de motifs généralisés [8]).

Motif	*Texte*	*occ*	*pos occ*
a#d#a	abracadadabra	17	3
ad#a	abracadadabra	5	3

Dans l'étude, le nombre de "gaps" (#) est un paramètre important lorsque l'on étudie le nombre d'occurrences.

Recherche d'expressions régulières. Tous les problèmes précédents se reformulent assez naturellement sous la forme de recherche de motifs décrits par une expression régulière. L'étude de ce dernier problème est donc de première importance.

Motif	*Texte*	*occ*	*pos occ*
$(ad)^+$	abracadadabra	3	2
$a(a+c+d)^+$	abracadadabra	12	6

La biologie moléculaire [27, 28, 29] fournit une source importante d'applications. Il existe de nombreux exemples, notamment car les recherches dans les séquences d'ADN doivent tenir compte de successions d'exons et d'introns, doivent chercher des signaux de départ et de fin des gènes, etc,.... Dans [30], les expressions régulières sont utilisées comme un modèle général de motif (qui permettent notamment de représenter les motifs dans le format PROSITE utilisés pour interroger les bases de données de séquences protéiques).

Pour chacun de ces types de motifs, des résultats asymptotiques ont été obtenus dans le cas de modèles probabilistes assez idéalisés (et assez éloignés de la réalité des séquences biologiques pour ne citer qu'elles).

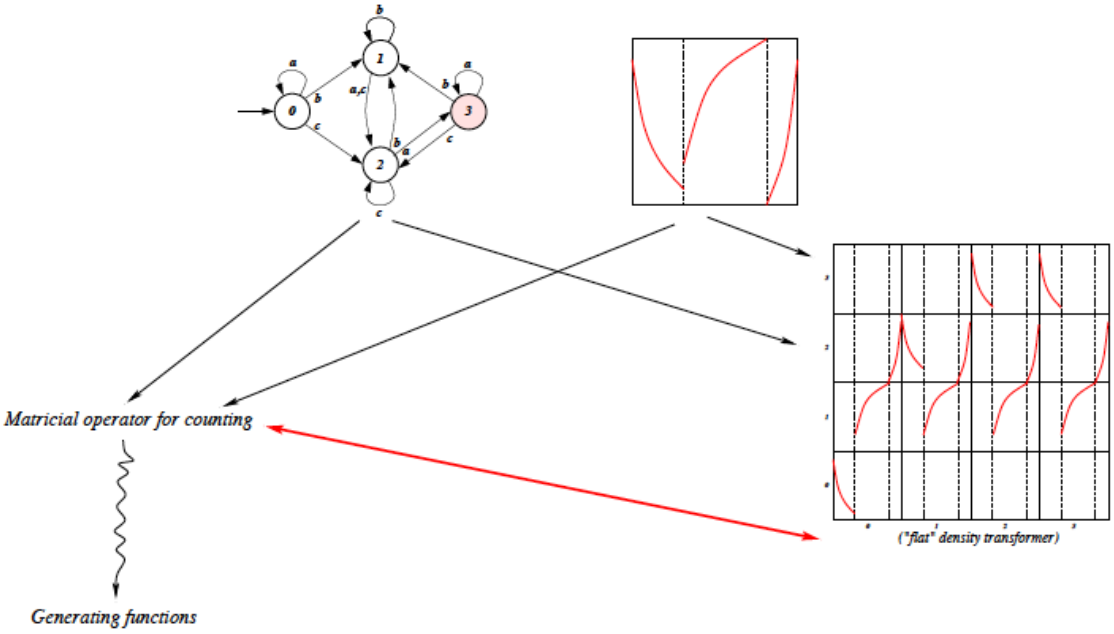
Les deux premiers problèmes, de recherche d'occurrences de mots ou d'ensemble de mots comme facteurs est étudié de manière intense depuis une trentaine d'années. Depuis [31, 32], le rôle des (auto)-corrélations a été montré comme très important. Des approximations normales [33, 34, 35], de Poisson [36, 37] ont été établies pour le nombre d'occurrences et des résultats de grandes déviations ont été montrés [38]. Le cas des motifs cachés a été abordé dans [39], dans le cas des sources sans mémoires. Enfin, le nombre de positions d'occurrences d'une expression régulière a été étudié, dans le cas des sources simples dans [40], le cas des chaînes de Markov étant abordé dans [41, 42] avec l'introduction d'algorithmes de calculs efficaces des probabilités d'occurrence.

Dans la suite, nous présentons succintement les résultats que nous avons obtenus dans [7] et qui concernent le nombre de positions d'occurrence d'une expression régulière dans le cas des sources dynamiques. Des résultats sur les autres problèmes de recherche de motifs dans le modèle des sources dynamique peuvent être trouvés dans [8], [43] ou encore [18].

3.2.2 Principes généraux de la méthode

L'idée majeure consiste à construire un opérateur qui combine à la fois l'opérateur de la source dynamique et l'automate qui représente le motif. De cet opérateur fonctionnel matriciel (et de ses objets propres dominants), il est possible de tirer des résultats asymptotiques sur le nombre de positions d'occurrence (une variable aléatoire qui peut être très simplement "encodée" dans cet opérateur). Nous verrons qu'il est nécessaire de passer par un opérateur intermédiaire (une version applatie de l'opérateur et conjuguée avec celui-ci). Ce dernier opérateur possède de bonnes propriétés en tant que système dynamique qui se transfèrent à l'opérateur matriciel.

Voici un schéma très succint de la méthode.



Soit $\mathcal{E}$ un motif donné sous forme d'une expression régulière. La première étape consiste à construire un automate associé au langage $\Sigma^*\mathcal{E}$ qui reconnaît l'ensemble des mots qui se terminent par une occurrence de $\mathcal{E}$. Dans la suite, on suppose que cet automate est fortement connexe et ce qui est écrit est vrai dans ce cadre. Le cas général est présenté dans [7]. Cet automate peut très facilement être transformé en un automate "marqué" qui compte les mots en "marquant" (i.e., en leur attribuant un coût de 1) toutes les transitions qui mènent à un état final et en rendant acceptants tous les états de l'automate. Ainsi, Tous les mots de Σ^* sont acceptés par l'automate. Pour obtenir le nombre de positions où une occurrence de $\mathcal{E}$ peut se terminer, il suffit de parcourir l'automate (en suivant les transitions étiquetées par le mot) et de compter le nombre de marques rencontrées qui est égal à ce nombre d'occurrences.

Pour construire l'opérateur générateur matriciel $\mathbb{T}(u)$ qui correspond à cet automate marqué (la variable u est là pour mémoriser le nombre de positions d'occurrences), il suffit de considérer la matrice de transition T de l'automate qui est telle que $T_{i,j}$ est l'ensemble des symboles qui permettent de passer de l'état i à l'état j. L'opérateur $\mathbb{T}(u)$ est tel que $\mathbb{T}_{i,j}(u) = u\sum_{m\in T_{i,j}} \mathbf{G}_{[m]}$, si j est un état final dans l'automate non marqué, et $\mathbb{T}_{i,j}(u) = \sum_{m\in T_{i,j}} \mathbf{G}_{[m]}$ sinon. Ici, $\mathbf{G}_{[m]}$ est l'opérateur générateur du symbole m de la source dynamique.

Cet opérateur permet d'obtenir une expression "manipulable" de la série génératrice double $M(z,u)$ qui permet de compter les positions d'occurrence de $\mathcal{E}$. En effet,

$$M(z,u) = \sum_{w\in\Sigma^*} p_w u^{C(w)} z^{|w|} = \int_0^1 (1,0,\ldots,0)\cdot(I - z\mathbb{T}(u))^{-1}\cdot(1,1,\ldots,1)[f](t)\,dt,$$

où f est la densité initiale. Ici, le quasi-inverse $(I - z\mathbb{T}(u))^{-1}$ permet de représenter toutes les trajectoires de longueurs quelconques, il suffit d'en extraire celles qui

commencent par l'état initial (le premier état ici) et qui se terminent dans n'importe quel état. On souhaite maintenant prouver que le théorème de Hwang s'applique à la série génératrice des moments de C_n, qui est fortement liée à $M(z,u)$ par la relation

$$\mathrm{E}\left[\exp\left(y\, C_n\right)\right] = [z^n]M(z, e^t),$$

et donc, on souhaite obtenir une décomposition de $M(z,u)$ (on encore de $\mathbb{T}(u)$). Malheureusement, le modèle matriciel d'opérateur générateur n'entre pas tout à fait dans le cadre des opérateurs associés aux bonnes sources dynamiques. Cependant, il est possible de définir un opérateur "plat" équivalent qui d'une part aura de bonnes propriétés dynamiques et d'autre part, sera lié à l'opérateur matriciel par une relation de conjugaison qui impliquera que les deux opérateurs possèdent le même spectre (et donc la même valeur propre dominante $\lambda(u)$), menant ainsi à la décomposition attendue pour $M(z,u)$.

3.2.3 Résultats

Nous présentons maintenant le résultat principal.

Théorème 3 *Soit $\mathcal{S}$ une "bonne" source dynamique.*
(i) Soit $\mathcal{E}$ une expression régulière amenant à un automate fortement connexe. Le nombre de positions d'occurrences de $\mathcal{E}$ dans un mot de longueur n construit par $\mathcal{S}$, noté $C_n(\mathcal{E})$, suit asymptotiquement une loi gaussienne dont le triplet caractéristique est donné par $r[C_n(\mathcal{E})] = O(1/\sqrt{n})$,

$$\mathrm{E}\left[C_n(\mathcal{E})\right] = \gamma_{\mathcal{E}} \cdot n + \gamma'_{\mathcal{E}} + O(\mu_{\mathcal{E}}^n),$$

$$\mathbf{Var}(C_n(\mathcal{E})) = \nu_{\mathcal{E}} \cdot n + \nu'_{\mathcal{E}} + O(\mu_{\mathcal{E}}^n).$$

Les constantes $\gamma_{\mathcal{E}}$ et $\nu_{\mathcal{E}}$ s'expriment à l'aide de la pression $\Lambda(t)$ de l'operateur $\mathbb{T}(e^t)$, précisément $\gamma_{\mathcal{E}} = \Lambda'(0)$, $\nu_{\mathcal{E}} = \Lambda''(0)$, tandis que $\mu_{\mathcal{E}} < 1$ est n'importe quel nombre réel strictement plus grand que le module de la valeur propre sous-dominante de $\mathbb{R}$.

3.2.4 Conclusion et perspectives

Dans ce travail, nous nous sommes restreint au cas où l'expression est "simple" dans le sens où même si l'automate (minimal) qui la représente n'est pas fortement connexe, il possède une unique composante fortement connexe qui va "capter" toute l'asymptotique. Ces résultats peuvent toutefois s'étendre au cas plus général où il y a plusieurs composantes fortement connexes finales. Dans ce dernier cas, toutes les composantes fortement connexes jouent un rôle dans l'asymptotique via leur valeurs propres dominantes, avec un poids plus important pour les valeurs propres "super-dominantes" (i.e., celles qui dominent toutes les autres).

3.3 Statistiques sur les oracles des facteurs

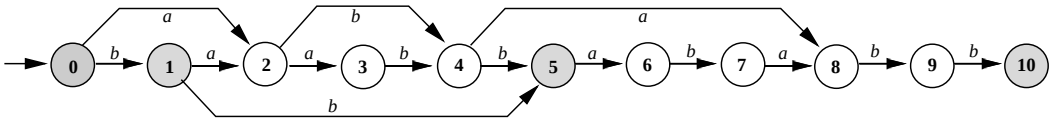
Les oracles des facteurs et des suffixes ont été introduits dans [44] afin de fournir une solution efficace et économe pour stocker tous les facteurs et/ou tous les suffixes d'un texte donné. Des estimations de la taille de ces oracles existent dans le pire des cas, nous proposons ici des résultats d'analyse en moyenne concernant la taille d'un oracle des facteurs/suffixes lorsque celui-ci a été construit en utilisant des mots de taille n produits par une source aléatoire binaire uniforme et sans mémoire. Pour cela, un nouvel oracle est défini. Il partage de nombreuses propriétés de l'oracle classique mais est plus simple à étudier et surtout, il fournit une borne supérieure pour la taille des oracles standards. Ce nouvel outil peut certainement être réutilisé pour étudier d'autres paramètres que la taille ou dans des généralisations de cette étude (à d'autres sources de mots par exemple).

3.3.1 min-Oracles et short-oracles des facteurs

min-Oracles Soit $w = w_1 w_2 \dots w_n$ une séquence de longueur $|w| = n$ sur un alphabet fini Σ. Soient i, j deux entiers tels que, $1 \leq i, j \leq n$, le mot $w[i \dots j] = w_i w_{i+1} \dots w_j$ est appelé un *facteur* de w (notons que lorsque $j < i$ le facteur résultant est le mot vide ε par convention). Un *suffixe* de w est un facteur de w dont une de ses occurrences dans w se termine en position n. Le i-ème suffixe de w, noté $Suff_w(i)$, est le suffixe $w[i \dots n]$ et il est de longueur $n+1-i$. Un *préfixe* de w est un facteur de w dont une de ses occurrences commence en position 1. Le i-ème préfixe de w, noté $Pref_w(i)$, est le préfixe $w[1 \dots i]$. Par convention, le mot vide ε est à la fois un préfixe et un suffixe de w. On dit qu'un suffixe de w est *maximal* s'il n'est pas égal à w et n'est pas le préfixe d'un autre suffixe de w. On dit qu'un suffixe w est *répété* si c'est un facteur de $w[1 \dots n-1]$. Dans le cas contrainre, on dit qu'il est *non répété*. Il est clair que les suffixes maximaux sont toujours des suffixes non répétés. L'inverse n'est vrai que lorsque les suffixes non-répétés dits propres (i.e., différents de w).

L'*oracle des facteurs/suffixes* de w est un automate déterministe qui possède $n+1$ états notés $0, 1, 2, \dots, n$, une *transition interne* $(i, w_{i+1}, i+1)$ pour tout état i distinct de n et au plus $n-1$ *transitions externes* notées (i, w_j, j), pour certaines paires i, j telles que $i+1 < j$. L'oracle des facteurs/suffixes est un automate homogène, toutes les transitions que arrivent au même état ont la même étiquette (gain en complexité spaciale). Chaque état de l'oracle des facteurs est final alors que seulement les états de l'oracle des suffixes qui reconnaissent un suffixe (sauf le mot vide) sont finaux (la figure 3.3.1 correspond à l'oracle des suffixes du mot $w = baabbababb$).

FIGURE 3.1 – Le min-oracle des suffixes $Omin(w)$ pour $w = baabbababb$. Les états finaux sont grisés.



Les oracles des facteurs/suffixes ont été introduits dans [44]. Ils peuvent être construits en utilisant un algorithme de construction en ligne de complexité linéaire. L'algorithme **Build_Oracle** donné ici (également proposé dans [44]) est quadratique, mais plus intuitif. Il va servir de base à la construction d'une autre version de l'oracle, plus adaptée à une étude en moyenne. Dans l'algorithme, $Omin(w)$ est indifféremment l'oracle des facteurs ou des suffixes.

L'oracle des facteurs ou des suffixes est une structure d'index approchée dans le sens ou certains mots, qui ne sont pas des facteurs peuvent tout de même être reconnus par l'automate. La figure 3.3.1 illustre cela. En effet $baabb$ n'est pas un suffixe de $w = baabbababb$ mais il est accepté par l'automate (état final 4). Ces mots sont appelés des intrus (ou encore *by-products*). Les compter (en moyenne) reste encore une question ouverte.

Algorithme Build_Oracle [44]

Entrée : Séquence w.

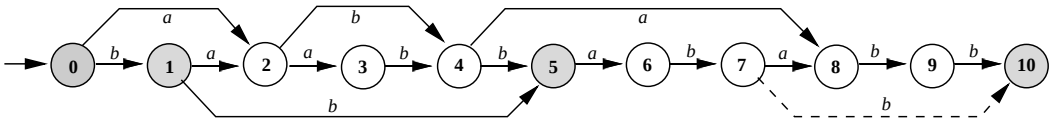
Sortie : $Omin(w)$.

1. **pour** i **allant de** 0 **à** n **faire**
2. créer un nouvel état i ;
3. **pour** i **allant de** 0 **à** $n-1$ **faire**
4. construire une transition de i à $i+1$ étiquetée par w_{i+1} ;
5. **pour** i **allant de** 0 **à** $n-1$ **faire**
6. soit x le plus petit mot reconnu à l'état i ;
7. **pour** tout $\gamma \in \Sigma$, $\gamma \neq w_{i+1}$ **faire**
8. **si** $x\gamma$ est un facteur de $w' = w[i - |x| + 1 \dots n]$ **alors**
9. soit j la position de fin de la première occurrence de $x\gamma$ dans w' ;
10. construire une transition de i à j étiquetée par γ
11. **fin si**
12. **fin pour**
13. **fin pour**

short-Oracle Soit u un mot sur l'alphabet Σ, on note $short(u)$ le plus court suffixe non répété de u (par convention, $short(\varepsilon) = \varepsilon$). On considère maintenant l'algorithme générique Build_Oracle dans lequel le générateur est maintenant la fonction $short()$. Ou de manière équivalente, en remplaçant l'étape 6 par $x = short(\mathit{Pref}_i(w))$, à la place de $x = min(\mathit{Pref}_i(w))$. L'automate obtenu par ce nouvel algorithme est noté $Oshort(w)$ et appelé le **short-oracle** of w. Ses versions facteurs et suffixes sont obtenues comme dans le cas du min-oracle.

Remarquons que pour certaines séquences w, $Omin(w)$ et $Oshort(w)$ sont identiques mais ce n'est pas toujours le cas puisque $short(u)$ et $min(u)$ peuvent être différents comme par exemple dans le cas où $u = baabbab$: $short(u) = bab$ et $min(u) = bbab$. Alors $Oshort(baabbababb)$ possède une transition externe étiquetée par b qui quitte l'état 7 (cf Figure 3.2) à cause de l'occurrence de $babb$ qui se termine à l'état 10. Au contraire, $Omin(baabbababb)$ n'a pas cette transition car $bbabb$ n'a pas d'occurrence qui se termine en un état $j > 7$.

FIGURE 3.2 – Le short-oracle des suffixes $Oshort(w)$ pour $w = baabbababb$. Les états finaux sont gris. La transition supplémentaire par rapport à $Omin(w)$ est dessinée en pointillés.



Comparaison des deux formes d'oracles Il est important de noter que bien que les transitions externes des min- et des short-oracles soient construites selon des règles proches et satisfont ainsi des propriétés similaires, il est beaucoup plus facile de calculer $short(u)$ que $min(u)$. En effet, $short(u)$ est obtenu simplement en considérant chaque suffixe de u et en testant s'il apparaît ailleurs dans u, tandis que pour calculer $min(u)$ il est nécessaire de construire le min-oracle (au moins en partie). En conséquence du fait qu'il suffit d'avoir des propriétés de nombre d'occurrences pour calculer $short(u)$, il est beaucoup plus facile d'estimer le nombre de transitions externes de $Oshort(w)$ que de $Omin(w)$. Cependant, les deux oracles sont liés et la propriété qui suit est essentielle dans l'étude.

Fait 1 *Soit w un mot et soient $ETmin(w)$ et $ETshort(w)$ les nombres respectifs de transitions externes respectivement de $Omin(w)$ et $Oshort(w)$. Alors on a $ETmin(w) \leq ETshort(w)$.*

3.3.2 Une Borne supérieure pour l'occupation mémoire d'un oracle

L'occupation mémoire d'un min-oracle construit à partir de w est la somme de trois quantités : le nombre d'états de $Omin(w)$ (fixé et égal à $n+1$), le nombre de transitions internes (fixé et égal à n) et le nombre de transitions externes.

Nous fournissons ici une estimation du nombre moyen de transitions externes d'un oracle $\mathrm{E}\,[ETshort_n]$, où $ETshort(w)$ compte le nombre de transitions externes de $Oshort(w)$ et $ETshort_n$ est sa restriction à $\mathbb{B}_n$. Cette estimation se calcule en temps linéaire.

Théorème 4 *Sous le modèle probabiliste $\mathbb{B}_n$ (i.e., l'ensemble des mots binaires aléatoires de taille n uniformément et identiquement distribués), l'occupation moyenne $\mathrm{E}\,[ETshort_n]$, en terme de transitions externes d'un short-oracle pour un mot de taille n vérifie*

$$\begin{aligned}\mathrm{E}\,[ETshort_n] &= pmin_0 + pmin_1 + (n-3) - \sum_{k=2}^{n-2} \frac{\gamma_{k+1}^{k-1}\lambda_{k+1}^{k-n} - \gamma_{k+1}^{n-2}\lambda_{k+1}^{-1}}{1-\gamma_{k+1}\lambda_{k+1}} \\ &+ \sum_{k=2}^{n-2} \frac{\gamma_k^{k-1}\lambda_{k+1}^{k-n} - \gamma_k^{n-2}\lambda_{k+1}^{-1}}{1-\gamma_k\lambda_{k+1}} + C_n + O(1)\end{aligned}$$

avec $\gamma_k = 1 - \frac{1}{2^k}$, $\lambda_k = 1 + \frac{1}{2^k}$ et $C_n = \frac{64}{3}\left(\frac{3}{4}\right)^n - \frac{16}{3}\left(\frac{1}{2}\right)^n - \frac{50}{3}\left(\frac{4}{5}\right)^n$.

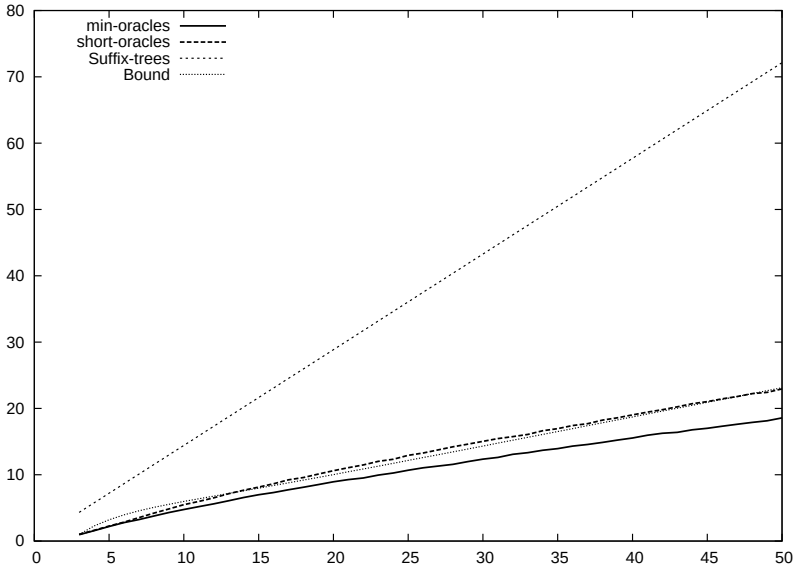
Ce théorème fournit le terme d'ordre principal de $\mathrm{E}\,[ETshort_n]$. Ensuite, le lemme 1 prouve que pour tout n, $\mathrm{E}\,[ETmin_n] \leq \mathrm{E}\,[ETshort_n]$. On obtient ainsi une borne asymptotique supérieure (une borne valide quand n est suffisamment grand) pour (le terme principal de) $\mathrm{E}\,[ETmin_n]$, le nombre moyen de transitions externes d'un min-oracle. Le tableau qui suit compare les valeurs de $\mathrm{E}\,[ETmin_n]$ avec cette borne supérieure. On y observe que la borne est valide dès $n \geq 4$.

n	3	4	5	6	7	8	9	10	11	12	13	14	15
$\mathrm{E}\,[ETmin_n]$	1	1.6	2.2	2.7	3.3	3.8	4.3	4.8	5.2	5.7	6.1	6.5	6.9
Bound	0.8	1.9	2.6	3.3	3.9	4.4	4.9	5.4	5.8	6.3	6.8	7.2	7.7

3.3.3 Conclusion et perspectives

Le résultat obtenu dans cette partie concerne une approximation de la probabilité qu'une transition externe existe dans un min- ou un short-oracle. Cette approximation permet d'étudier l'occupation mémoire moyenne de ces oracles. Le but principal étant de comparer cette occupation mémoire avec celles d'autres structures d'index comme les arbres des suffixes dont l'occupation en mémoire dépend de son nombre d'arêtes internes et connu comme étant en moyenne d'ordre $n/\log 2$ (voir [45] pour plus de détails). La figure 3.3 compare la borne que nous avons obtenu avec ce nombre. Nous avons aussi tracé une moyenne empirique du nombre réel de transitions externes dans les min-oracles et les short-oracles pour voir de combien notre borne s'en éloigne.

FIGURE 3.3 – Comparaison de l'occupation mémoire des short-oracles, min-oracles et arbres des suffixes



Il reste à noter qu'une des principales questions encore ouverte à ce jour sur cette structure d'oracle concerne le nombre de mots, reconnus par un oracle mais

qui ne sont pas des facteurs ou des suffixes du mot qui a permis de construire l'oracle. Ces mots sont parfois appelés "intrus" ou "by-products". Ce sont juste des mots que l'on ne souhaiterait pas voir stocker par la structure d'index. Le résultat que nous avons obtenu peut être très utile pour attaquer ce problème. En effet, le nombre total de mots reconnus par un oracle des facteurs s'exprime comme la somme $\sum_{k=0}^{n} N_k$, où N_i est le nombre moyen de mots reconnus à l'état i. Ces nombres N_k vérifient la récurrence suivante, qui fait intervenir également la probabilité qu'une transition existe entre l'état i et l'état j : $N_0 = 1$ et pour tout $0 < j \leq n$, $N_j = \sum_{i=0}^{j} pmin_{i \to j} N_i$. Cela reste un challenge de résoudre cette récurrence. Néanmoins, il est assez facile d'en obtenir un algorithme de programmation dynamique qui calcule itérativement les nombres N_j en utilisant la formule que nous avons obtenue pour $pmin_{i \to j}$. On peut donc obtenir une borne supérieure pour le nombre moyen de mots reconnus par un min-oracle de la même manière que la borne obtenue pour le nombre de transitions externes.

4

Comprendre, analyser, manipuler une source de mots : bio-informatique des systèmes.

Le concept de mot rejoint la biologie des systèmes lorsque l'on considère des encodages des trajectoires par des mots du système vivant (ou de quantités liées au système vivant). Alors, comprendre le système vivant revient à comprendre et appréhender le processus qui a permis de produire ces mots-trajectoires. Les sources sont alors complexes et deviennent l'objet d'étude.

Après une brève introduction à quelques problématiques de l'étude de systèmes biologiques, nous présentons deux applications précises. Premièrement, nous étudierons l'impact du temps sur la dynamique du système. Nous verrons que lorsque l'on parle du temps, il y a deux notions : un temps chronologique (comment s'enchaînent les évènements) et un temps chronométrique (combien de temps dure un évènement). Chacun de ces aspects sera abordé dans une courte partie. Ensuite, nous présentons une manière assez générale d'ajouter des aspects quantitatifs à des modèles qui représentent des évènements biologiques en vue de confronter un modèle de stratégie individuelle à des données typiquement mesurées sur une population d'individus. Cette partie se termine par une très rapide présentation de travaux en cours et qui adressent des problématiques de reconstruction de réseaux biologiques. La reconstruction (semi-)automatique de tels réseaux est un des enjeux majeurs actuellement en biologie des systèmes.

4.1 Biologie des systèmes : un focus sur les réseaux de régulation de gènes

La modélisation en biologie des systèmes est un problème à multiples facettes qui dépend de l'angle d'observation adopté sur le système. Cependant, de tous

points de vue, on observe des interactions entre les différents composants de ce système complexe. Dans cette partie, je me focaliserai sur un type de composant, le gène, qui par le biais d'enchaînements de réactions complexes qui passent par la création de protéines par exemple, pour avoir des effets sur d'autres gènes. L'ensemble de ces interactions constitue ce qu'on appelle les *réseaux de régulation de gènes*.

4.1.1 Les réseaux de régulation de gènes : une définition

Les réseaux de régulation de gènes (RRG) représentent les interactions fonctionnelles entre différents composants macromoléculaires comme l'ADN ou les protéines. Lorsque ces interactions sont simples et ne font pas intervenir plus de deux composants en même temps, un RRG est typiquement décrit par un graphe orienté dont les sommets sont les composants et dont les arêtes ont non seulement une orientation (action d'un composant sur un autre) mais aussi très souvent un type (eg. l'action est une inhibition) (cf la figure 4.13 (A)).

De plus, un gène peut contribuer à l'activation ou l'inhibition d'un autre gène via la création de complexes protéiques. Cet effet, résultat de l'action conjointe de plusieurs gènes est appelé co-activation ou co-inhibition. L'action n'est alors réalisée que lorsque tous les gènes nécessaires sont ou ont par le passé été actifs. Pour aller encore plus loin, la régulation des gènes est encore plus complexe car elle peut faire intervenir des concentrations de protéines, ou des niveaux de concentration de gènes (en terme de concentration en ARN messagers).

Parce qu'une telle complexité ne peut pas être représentée fidèlement, un des points clé dans l'étude de réseaux de régulation va consister à choisir le formalisme de modélisation adapté à la question posée par le biologiste. Nous présentons maintenant quelques uns de ces formalismes.

4.1.2 Les réseaux de régulation de gènes : formalismes

De nombreuses compilations présentent les formalismes adoptés pour modéliser les réseaux de régulation géniques [46, 47, 48, 49, 50].

Lorsque l'on parle de modélisation de systèmes complexes, un des premiers formalismes consiste en une modélisation par u système d'équations différentielles. C'est le plus couramment utilisé pour décrire *quantitativement* la cinétique d'un système dynamique. La cinétique de la régulation d'un réseau de gènes est alors modélisée par un ensemble d'équations exprimant le taux de production de chaque composant du système (ARN, protéine, autre molécule) comme une fonction des concentrations des autres composants. Mais la non-linéarité des fonctions utilisées rend ce type de formalisme réfractaire à l'analyse mathématique. Le seul recours qui reste pour identifier des propriétés du système étudié est alors la simulation numérique. De telles simulations, opérées en variant divers paramètres, permettent de mettre en relation des comportements du système étudié et des états physiologiques connus de la cellule, ainsi que des états non encore observés expérimentalement. Cependant, *in vivo* comme *in vitro*, la difficulté à obtenir des mesures permettant d'étalonner ce type de modèle limite leur utilisation à quelques cas très bien étu-

diés. Des modélisations plus simples, basées sur des équations linéaires par morceaux, ont été proposées pour palier à cette difficulté d'analyse [51]. Même si ces modélisations réduisent le nombre de paramètres nécessaires, elle reste encore limitées à des systèmes comportant peu de composants. Ces méthodes de modélisation sont qualifiées de modélisations quantitatives (basées sur les quantités exactes des composants) ou semi-quantitaves.

Pour tenter d'identifier des classes de comportement (ou trajectoires) du système modélisé, correspondant à des domaines de valeurs prises par les paramètres du modèle, il est intéressant d'adopter une approche plus discrète. Ce type de modélisation permet d'échapper à la nécessité de mesures fines et de s'en tenir à des niveaux d'expression de gènes. Les modélisations sont alors dites qualitatives, basées sur des propriétés vérifiées par les composants du modèle (un gène est actif, ou un gène est "plus" actif qu'un autre, etc,...). La modélisation par réseau booléen [52, 53, 54] associe à chaque gène une variable désignant son état (actif/inactif). A l'instant $t + 1$, l'état d'un gène est modélisé comme une fonction booléenne des états à l'instant t, des gènes activateurs ou inhibiteurs dont il dépend. Grossièrement, chaque gène peut-être actif ou inactif. Un état du système à l'instant t est alors représenté par un vecteur booléen X_t. Les activités des gènes se traduisent par des productions de protéines qui vont à leur tour activer ou inhiber d'autres gènes, induisant à un instant suivant un autre état du système. Il a été montré qu'on pouvait associer des phénotypes particuliers à certains attracteurs du système. Etudier la dynamique du système va donc consister à déterminer ces attracteurs, comprendre comment on peut arriver à ces attracteurs (dynamique transitoire) et comment contrôler le système pour favoriser ou non certains attracteurs (phénotypes). Kauffman [55] a proposé d'utiliser un modèle simple du type $X_t \rightarrow f(X_t)$, où f est une fonction vectorielle booléenne pour décrire cette dynamique. Le constat a très vite été fait que ce modèle est trop pauvre pour représenter les modèles biologiques complexes. Depuis, deux branches ont émergé et évolué de manière indépendante, se basant chacune sur une famille d'extensions différentes des réseaux booléens. Une branche "européenne", portée par René Thomas [56], va s'employer à développer des méthodes de vérification formelles de propriétés qualitatives de la dynamique du système, autour d'un modèle enrichi par de l'indéterminisme (plusieurs transitions sont possibles à partir d'un état, leur choix est indéterministe). La sémantique formelle du modèle a également été enrichie pour rendre encore plus compte des particularités des systèmes biologiques (domaines discrets mais plus forcément booléens, ajout de délais de réaction, notions de synchronisation complexes entre gènes...). Ces derniers modèles sont basés sur une vision qualitative et abstraite des données. Ils ne peuvent être construits que par des experts en modélisation. Une autre branche a été initiée en 2002 par une communauté plutôt américaine. Son but est de partir des données "brutes" et de construire des modèles. Les modèles sont probabilistes, appelés Probabilistic Boolean Networks [57, 58, 59], pour tenir compte de bruits et autres phénomènes complexes cachés dans les données. Les axes de recherche principaux de cette thématique sont la reconstruction de réseaux et l'étude de distribution stationnaires de modèles probabilistes de réseaux booléens. Ils sont construits de manière automatique sans réelle étude de robustesse vis-à-vis d'autres données. Bien qu'éloignées dans les approches informatiques développées,

ces deux branches (vérification de modèles sur réseaux multivalués et étude de Probabilistic Boolean Networks) manipulent des modèles relativement proches et des questions naturelles se posent lorsque l'on compare les deux approches.

Les travaux développés dans la suite concernent ces modèles qualitatifs. Nous verrons qu'ils peuvent être raffinés de plusieurs manières, en ajoutant des contraintes temporelles d'une part, puis en ajoutant des contraintes liées aux quantités de produits présents dans le système.

4.2 Réseaux de gènes : influence du temps *chronologique* sur la source

Dans cette partie, nous montrons que des phénomènes de priorités entre gènes peuvent avoir une influence importante sur le réseau, et donc sur ses objets dominants. Ces résultats sont illustrés pour mettre en évidence les différences de distributions limites dans les Probabilistic Boolean Networks, lesquels sont de manière évidente des sources dynamiques paramétrables.

4.2.1 Probabilistic Boolean Networks : une définition

Le réseau booléen (en anglais non traduit dans la suite du manuscript "Boolean Network") est le cœur du Probabilistic Boolean Network. Il est donc important d'en avoir une définition avant de parler de son extension probabiliste.

Boolean Networks

Les BN ont été introduits dans les années 60 par Kauffman [60, 55]. Ils sont très rapidement devenus un objet d'intérêt central tant en physique qu'en biologie. Leur application en biologie des systèmes se focalise sur l'étude des réseaux de gènes. Dans ce cas, les gènes sont assimilés à des interrupteurs qui peuvent être dans deux états : ouvert ou fermé. Cette hypothèse est une simplification d'activation d'un gène sur un autre par une fonction en marche d'escalier. Les interactions entre les gènes permettent de construire un réseau. Bien sûr, l'évolution de l'activité d'un gène dépend des activités d'autres gènes. Le but d'un BN est de trouver une abstraction manipulable permettant de représenter les évolutions des activités des gènes et d'en tirer des propriétés émergentes.

Le BN est défini de la manière suivante :

Définition 3 *Un Boolean Network $B = (V, F)$ est une paire telle que*
- $V = \{x_1, \ldots, x_n\}$ *est un ensemble de variables booléennes (les gènes),* $\forall i, x_i \in \{0, 1\}$.
- $F = \{f_1, \ldots, f_n\}$ *est un ensemble de fonctions booléennes,* $\forall i, f_i : \{0, 1\}^n \to \{0, 1\}$. *Ici, f_i décrit l'évolution du gène i (en fonction des autres gènes).*

Ces réseaux permettent de décrire la dynamique d'un phénomène. Formellement, si $X(t) = (x_1(t), \ldots, x_n(t))$ est le vecteur donnant la valeur de toutes les variables au temps t, alors $X(t+1) = (x_1(t+1), \ldots, x_n(t+1))$, où $\forall i, x_i(t+1) =$

$f_i(x_1(t), \ldots, x_n(t))$ donne la valeur de toutes les variables après une itération du BN. Il faut noter que pour n gènes, le nombre possible de BN est 2^{n2^n}. Parmi eux, seuls quelques-uns ont un réel sens biologique. Leur identification est une des question majeure en théorie des Boolean Networks.

Cette dynamique est généralement représentée par un graphe, appelé *graphe dynamique* ou encore *graphe d'états*. Il est défini par

Définition 4 *Soit $B = (V, F)$ un BN et $n = |V|$. Le graphe dynamique $\mathcal{G} = (\{0,1\}^n, E)$ associé à B est un graphe orienté qui possède une arête de $(x_1, \ldots, x_n)$ à $(x'_1, \ldots, x'_n)$ si et seulement si pour tout $i = 1, \ldots, n$, $x'_i = f_i(x_1, \ldots, x_n)$.*

Dans la suite, nous montrons que la topologie du graphe est affectée par les effets de priorités et de coopération entre gènes.

Ces graphes dynamiques sont importants car ils mettent en évidence les attracteurs du système dont il a été montré que, dans le cadre de l'étude des réseaux de gènes, ils pouvaient être associés à certains phénotypes [61, 62]. Comme leur étude est importante, plusieurs approches ont été développées : des simulations de la distribution stationnaire ; des algorithmes de théorie des graphes pour en étudier la topologie, des approches de vérification de modèles,...

Probabilistic Boolean Networks

Les Boolean Networks ne sont pas toujours représentatifs du comportement "correct" de modèles biologiques complexes. En fait, les BN manquent de flexibilité pour prendre en compte la complexité inhérente des interactions entre gènes (par exemple, la non-linéarité due aux régulations post-transcriptionnelles) ou encore pour refléter des données erronées ou incomplètes. Pour prendre en compte cette flexibilité, Shmulevich et ses co-auteurs [59] ont introduit les réseaux booléens probabilistes (Probabilistic Boolean Networks (PBN)), qui sont une extension probabiliste des BN.

Définition 5 *Un Probabilistic Boolean Network $B = (V, \mathcal{F})$ est un couple où*

- *$V = \{x_1, \ldots, x_n\}$ est un ensemble de variables booléennes (i.e. les gènes), $\forall i, x_i \in \{0,1\}$;*
- *$\mathcal{F} = \{F_1, \ldots, F_n\}$ est un ensemble où*

$$F_i = \{(f_i^{(1)}, p_i^{(1)}), \ldots, (f_i^{(l_i)}, p_i^{(l_i)})\}$$

est un ensemble de paires composées par une fonction booléenne et une probabilité. Pour tout i, on a $\sum_{k\in\{1,\ldots,l_i\}} p_i^{(k)} = 1$. Ici, l'évolution du gène i est prédite par $f_i^{(k)}$ avec probabilité $p_i^{(k)}$.

La dynamique du système est maintenant décrite en utilisant des vecteurs de variables aléatoires booléennes. $(X_1(t), \ldots, X_n(t))$, telles que $\forall i, \forall b \in \{0,1\}$,

$$\text{Prob}\{X_i(t+1) = b\} = \sum_{k\in\{1,\ldots,l_i\}, f_i^{(k)}(X_1(t),\ldots,X_n(t))=b} p_i^{(k)}.$$

Cela ajoute une composante d'incertitude au graphe dynamique.

Il faut noter que le graphe dynamique est maintenant une chaîne de Markov dont les probabilités de transition sont définies par des combinaisons des $p_i^{(k)}$. En effet, la transition de l'état $S = (x_1, \ldots, x_n)$ à l'état $S' = (x'_1, \ldots, x'_n)$ a pour probabilité

$$p_{S \to S'} = \sum_{(k_1,\ldots,k_n) \in U(S,S')} \prod_{j=1}^{n} p_j^{(k_j)}, \tag{4.1}$$

où $U(S, S') \subset \prod_{i=1,\ldots,n} \{1, \ldots, l_i\}$ est un ensemble contenant toutes les combinaisons de fonctions booléennes qui permettent de passer de S à S'. Il est parfaitement défini de la manière suivante, for all $(k_1, \ldots, k_n) \in U(S, S')$,

$$S' = (f_1^{(k_1)}(S), \ldots, f_n^{(k_n)}(S)).$$

Nous montrons maintenant que la stratégie de mise à jour a un effet sur l'ensemble $U(S, S')$, donc sur les probabilités de la chaîne de Markov.

4.2.2 Stratégies de mise à jour

Nous proposons ici plusieurs stratégies pour mettre à jour le systèmes biologique. Ces stratégies ont un impact sur le graphe dynamique. C'est un problème relativement classique en informatique qui revêt un caractère important dans le contexte de la biologie des systèmes puisque cette stratégie peut-être associée à des propriétés biologiques.

Mise à jour synchrone

C'est la stratégie de manière classique utilisée dans le contexte des réseaux booléens. De manière intuitive, elles correspondent à des successions d'observations de l'activité des gènes à des temps fixés. Dans cette stratégie, on suppose que toutes les modifications des activités des gènes sont effectuées en même temps. Si $(x_1(t), \ldots, x_n(t))$ est l'état booléen de tous les gènes au temps t, alors $(x_1(t+1), \ldots, x_n(t+1))$, où pour tout i, $x_i(t+1) = f_i(x_1(t), \ldots, x_n(t))$, est l'état booléen de tous les gènes au temps $t + 1$.

Mise à jour asynchrone

On peut observer qu'il est très rare que deux gènes indépendants aient des modifications dans leur activité exactement en même temps. Ainsi, si on voit la dynamique booléenne comme une discrétisation de la dynamique continue, la mise à jour synchrone n'est pas réaliste. Ici, on suppose que seul un gène peut avoir une modification de son activité. Cette stratégie a été appliquée à d'autres modèles discrets de représentation de la dynamique comme les Random Boolean Networks par Deng et al. [63] et pour l'étude des modèles de Thomas [64, 65] qui sont des extensions discrètes et non déterministes des Boolean Networks. La stratégie asynchrone se décrit par : si $(x_1(t), \ldots, x_n(t))$ est le vecteur des activités des gènes au temps t, il est nécessairement suivi par un parmi les n états possibles $(x_1(t+1), \ldots, x_n(t+1))$ où $x_i(t + 1) = f_i(x_1(t), \ldots, x_n(t))$ si $i = j$ et $x_i(t + 1) = x_i(t)$ sinon, pour tout $j \in \{1, \ldots n\}$.

Mise à jour complexe 1 : synchronisations entre gènes

Les systèmes biologiques ont souvent besoin de plasticité dans leur stratégies de mise à jour. En effet, certains gènes sont co-régulés, donc évoluent au même rythme, d'autres sont indépendants. Ainsi, l'hypothèse d'évolution asynchrone n'est plus réaliste à cause des spécificités des régulations et souvent également car la connaissance de ces régulations est incomplete. Dans ce contexte, Naldi *et al* [66] ont proposé une stratégie mixte qui combine des mises à jour synchrones pour les gènes qui ont une vitesse de régulation similaire et des mises à jour asynchrones pour les autres. Formellement, Soit $G = \{g_1, \ldots, g_m\}$, une partition disjointe de $\{1, \ldots, n\}$ composée d'ensembles non vides (*i.e.,* $\forall u, g_u \neq \emptyset$, $\bigcup_{u=1}^{m} g_u = \{1, \ldots, n\}$ et $\forall u \neq v,\ g_u \cap g_v = \emptyset$). Cette partition décrit les phénomènes de synchronisations entre gènes. Les deux cas extrêmes sont $G = \{\{1, \ldots, n\}\}$ qui correspond aux mises à jour synchrones et $G = \{\{1\}, \ldots, \{n\}\}$ correspondant aux mises à jour asynchrones. Enfin, si $(x_1(t), \ldots, x_n(t))$ est le vecteur d'état booléen des activités des gènes au temps t, il existe au plus m successeurs possibles $(x_1(t+1), \ldots, x_n(t+1))$ tels que pour tout $u \in \{1, \ldots, m\}$, on a $x_i(t+1) = f_i(x_1(t), \ldots, x_n(t))$ lorsque $i \in g_u$ et $x_i(t+1) = x_i(t)$ sinon.

Mise à jour complexe 2 : synchronisations entre fonctions

Il est parfois nécessaire d'être encore plus précis dans la notion de synchronisation. Dans cette partie, nous introduisons une notion de synchronisation entre fonctions. En effet, on peut considérer que les différentes fonctions booléennes alternatives d'un gène donné dans un PBN correspondent à des composants de réactions complexes. Il est donc intuitivement clair que toutes les fonctions qui correspondent à une même réaction complexe doivent être synchronisées. Ceci est réalisé en définissant des synchronisations au sein d'une partition des fonctions. Soit $S = \{s_1, \ldots, s_m\}$, une partition disjointe de $I = \bigcup_{i=1,\ldots,n} \{i\} \times \{1, \ldots, l_i\}$ composée d'ensembles non vides (*i.e.,* $\forall u, s_u \neq \emptyset$, $\bigcup_{u=1}^{m} s_u = I$, $\forall u \neq v,\ s_u \cap s_v = \emptyset$ et $\forall s \in S,\ \forall (g,k), (g',k') \in s,\ g = g' \Rightarrow k = k'$). Cette partition décrit les synchronisations entre les fonctions. Alors, si $(x_1(t), \ldots, x_n(t))$ correspond à l'état booléen de toutes les activités des gènes au temps t, il existe au plus m successeurs possibles $(x_1(t+1), \ldots, x_n(t+1))$ où pour tout $u \in \{1, \ldots, m\}$, $x_i(t+1) = f_i^{(k)}(x_1(t), \ldots, x_n(t))$ si $(i,k) \in s_u$ et $x_i(t+1) = x_i(t)$ sinon.

Interprétation probabiliste des synchronisations

De manière naturelle, ces stratégies de mise à jour, qui impactent l'interprétation du réseau booléen, jouent également un rôle majeur lorsque l'on s'intéresse aux réseaux booléens probabilistes. Nous illustrons ce point en montrant comment les stratégies de mise à jour mixtes s'interprètent dans le contexte des PBN. Ici, le graphe de la dynamique est une chaîne de Markov à 2^n états, dont la matrice de transition $T = (p_{S \to S'})_{S,S' \in (\{0,1\}^n)^2}$ est défini comme suit.

Il est facile d'étendre la définition de la matrice de transition (4.1) pour prendre en compte les effets de mise à jour. En effet, considérions un état S et un de ses successeurs possibles S' (dans le sens de ce qui a été défini à la section précédente, en

tenant compte de la synchronisation). L'ensemble $U(S, S') \subset \prod_{i=1,\dots,n}\{0, 1, \dots, l_i\}$ est maintenant composé des n-uplets où les zéros correspondent aux gènes qui ne sont pas modifiés par la mise à jour. Si on adopte la convention que $p_i^{(0)} = 1$, la définition (4.1) reste valide.

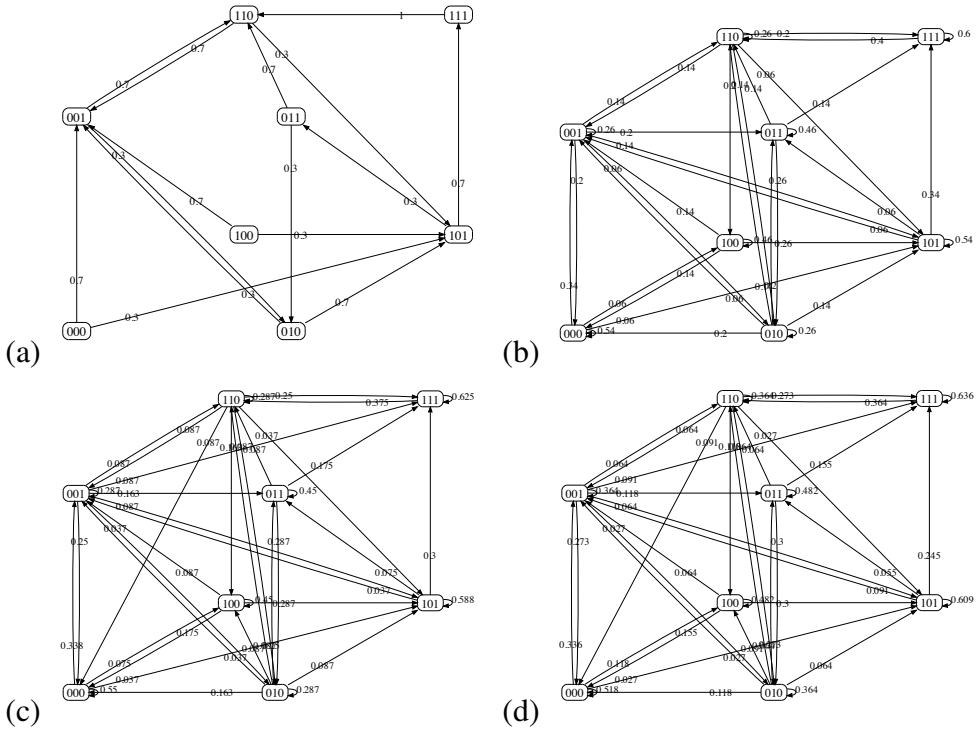
Illustration sur un exemple jouet

Considérons le PBN defini par :

$$\begin{aligned}
f_1^{(1)}(x_1, x_2, x_3) &= x_1 \wedge x_2 \vee \neg x_2 \wedge \neg x_3, & p_1^{(1)} &= 0.3\\
f_1^{(2)}(x_1, x_2, x_3) &= \neg x_1 \wedge x_2 \vee x_3, & p_1^{(2)} &= 0.7\\
f_2^{(1)}(x_1, x_2, x_3) &= x_1 \wedge x_3 \vee \neg x_1 \wedge x_2, & p_2^{(1)} &= 1\\
f_3^{(1)}(x_1, x_2, x_3) &= x_1 \wedge \neg x_2 \vee \neg x_3 & p_3^{(1)} &= 1
\end{aligned}$$

La figure 4.1 illustre différentes stratégies de mise à jour. Il faut noter que la stratégie de mise à jour affecte la forme du graphe dynamique. De plus, même si la forme du graphe dynamique n'est pas affectée, les probabilités de transition sont affectées.

FIGURE 4.1 – Graphe dynamique d'un PBN avec plusieurs stratégies de mise à jour : (a) mise à jour synchrone ; (b) mise à jour asynchrone ; (c) synchronisation entre gènes de partition $G = \{\{1, 2\}, \{3\}\}$; (d) synchronisation entre fonctions de partition $S = \{\{(1, 2), (2, 1)\}, \{(1, 1)\}, \{(3, 1)\}\}$.



4.2.3 Conclusion et perspectives

Ici, j'ai montré l'importance de la stratégie de mise à jour dans le processus de modélisation. Ces stratégies modifient le comportement dynamique du PBN. Des informations sur la stratégie la plus appropriée sont en général disponibles dans la littérature. Et si c'est le cas, une question qui reste importante va consister à mettre au point des algorithmes d'inférence de PBN qui permettent de tenir compte de ces mises à jour.

Une autre motivation importante pourrait être de comparer ces résultats avec les stratégies de mises à jour liées à d'autres modélisations discrètes. Par exemple, un autre modèle très utilisé dans l'étude des réseaux de régulation de gènes est le réseau de Thomas (cf [64, 65] pour plus de détails). Il est défini par le biais de points focaux (appelés plus communément des paramètres K du modèle).

Dans ces modèles, les comportements des gènes peuvent changer en fonction de leur niveau d'activation (*i.e.*, en fonction de la quantité d'ARNm dans la cellule), ce qui donne des réseaux dont les graphes d'états sont discrets.

Il est relativement simple de définir un équivalent discret aux PBN.

Définition 6 *Un Probabilistic Discrete Network (PDN) $B = (V, \mathcal{F})$ est défini par une paire où*

- $V = \{x_1, \ldots, x_n\}$ *est un ensemble de variables discrètes (i.e. les niveaux d'activation des gènes),* $\forall i, x_i \in D_i = \{0, \ldots, d_i\}$, d_i *est le niveau d'activation maximal du gène* i *;*
- $\mathcal{F} = \{F_1, \ldots, F_n\}$ *est un ensemble où*

$$F_i = \{(f_i^{(1)}, p_i^{(1)}), \ldots, (f_i^{(l_i)}, p_i^{(l_i)})\}$$

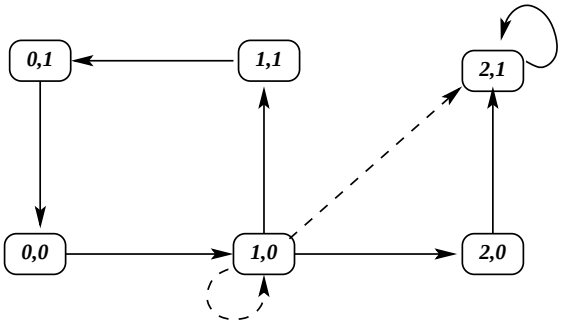
est un ensemble de paires composées d'une fonction discrète (de $\prod_j D_j$ dans D_i) et une probabilité qui vérifie pour tout i, $\sum_{k\in\{1,\ldots,l_i\}} p_i^{(k)} = 1$. Ici, L'évolution du gène i est prédite par $f_i^{(k)}$ avec probabilité $p_i^{(k)}$.

Il est aussi utile d'ajouter des hypothèses de mise à jour qui imposent que le niveau d'activation d'un gène ne peut pas augmenter ou diminuer de plus de 1. Cela revient à considérer une distance de Tchebychev $d_\infty(x, y) = \max_i |x_i - y_i|$ d'au plus 1 entre un état et ses successeurs possibles (rq. pour les réseaux de Thomas, une hypothèse similaire est aussi faite en utilisant la 1-distance $d_1(x, y) = \sum_i |x_i - y_i|$).

Afin de comparer ces deux formalismes, il est nécessaire d'imposer des synchronisations de fonctions dans le PDN. En effet, si l'on regarde la figure 4.2, la seule manière d'avoir une équivalence (au moins topologique) entre les deux graphes dynamiques, est d'imposer une synchronisation entre les fonctions $\{f_1^{(1)}, f_2^{(1)}\}$ et $\{f_1^{(2)}, f_2^{(2)}\}$.

Enfin, c'est un travail préliminaire qui, à mon avis, ouvre des pistes intéressantes pour envisager des méthodes d'inférences de probabilités sur les graphes d'états beaucoup plus fines que les méthodes basées sur des calculs empiriques (de type Baum-Welch). Ceci, grâce aux expressions analytiques obtenues pour les probabilités de transition du graphe d'état, fonction des probabilités inconnues des utilisations des fonctions booléennes, et en utilisant les méthodes d'inférences de

FIGURE 4.2 – le graphe dynamique d'un PDN (toutes les arêtes) et son équivalent de Thomas (seulement les arêtes pleines) et leur définition.



x	y	K	$f_1^{(1)}$	$f_1^{(2)}$	$f_2^{(1)}$	$f_2^{(2)}$
0	0	1,0	1	1	0	0
0	1	0,0	0	0	0	0
1	0	2,1	2	1	0	1
1	1	0,1	0	0	1	1
2	0	2,1	2	2	1	1
2	1	2,1	2	2	1	1

sources dynamiques pondérées qui sont décrites un peu plus tard dans ce manuscript. C'est un travail futur à mener et qui permettra de réconcilier les méthodes de construction de modèles à base de connaissances "expertes" (comme le sont les modèles de Thomas) et les modèles inférés à base de données (comme le sont les PBN).

4.3 Influence du temps *chronométrique*

Dans cette partie, nous abordons une autre notion temporelle, cette fois-ci d'ordre chronométrique. C'est un travail, réalisé en collaboration avec l'équipe MeForBio de l'IRCCyN, qui est le coeur de la thèse de Jamil Ahmad. J'en présente ici les principes généraux et donne les résultats obtenus lors de cette collaboration, publiés dans [67]. De manière plus précise, j'ai apporté à cette collaboration une méthode probabiliste de sélection des états d'intérêt sans laquelle toute l'étude n'aurait pas été possible pour des questions d'explosion combinatoire.

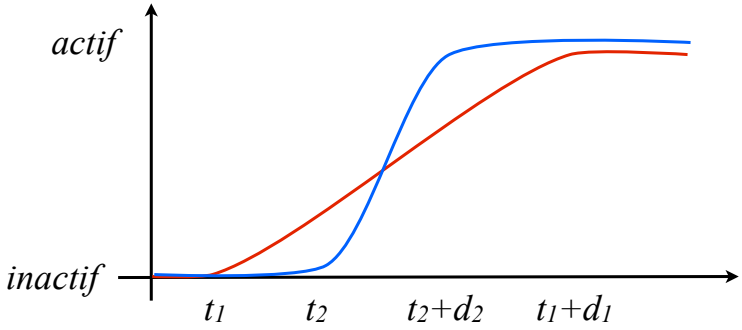
Plutôt que de considérer la chronologie des évènements, il est parfois utile d'introduire des mesures chronométriques des évènements (i.e., des durées d'évènements, des délais sur les transitions, etc,...). Il est clair que ces délais vont aussi jouer un rôle important dans la dynamique (un exemple intuitif est donnée par la figure 4.3).

Ces délais qui sont liés à des constantes cinétiques de réaction sont en général inconnus ou au moins mal connus, en particulier en condition in vivo, qui sont les conditions cibles en biologie des systèmes.

Nous avons abordé ce problème d'intégration de délais dans le cadre de la modélisation des réseaux de gènes (via une extension chronométrique au modèles de Thomas). Il en résulte un modèle réaliste de graphe de régulation de gènes, pour lequel des contraintes temporelles sur les délais de réaction sont ajoutées. Le but ultime est d'utiliser sur ce dernier modèle des techniques de vérification de modèles qui prend en compte tant des informations chronologiques que des informations chronométriques.

Cette méthode est illustrée sur le modèle classique *Escherichia coli* soumis à un

FIGURE 4.3 – Exemple : imaginons deux gènes g_1 et g_2 dont l'activité change à des temps respectifs t_1 et t_2 ($t_1 < t_2$) avec des délais différents ($d_1 > d_2$). Quel gène est activé le premier ? Comment prendre en compte les délais ?



stress de privation de ressource en carbone [68].

4.3.1 horloges, délais, automates temporisés : une définition

La méthode repose sur un modèle d'automate temporisé, intégrant des horloges et des contraintes sur ces horloges. Nous présentons ici le coeur de la méthode.

Horloges et délais

L'évolution de l'expression d'un gène donnée est généralement décrite par une fonction sigmoïdale (cf Figure 4.4).

Dans le formalisme développé par Thomas, Ceci est totalement ignoré puisque les transitions sont considérées comme instantanées, donc décrites par des fonctions en escalier (cf. Figure 4.4 (b)). En biologie des systèmes, plusieurs paradigmes ont été proposés pour simuler une évolution continue [69, 70, 71, 72]. Dans [73, 74], ce problème est abordé via un raffinement des modèles discrets pour lesquels l'évolution est instantanée. Ici, l'évolution sigmoïdale est approximée par une fonction linéaire par morceau (cf Figure 4.4 (c)). Ceci introduit une notion de *délais* nécessaire pour qu'un gène passe du niveau d'expression a au niveau d'expression $a+1$ ou $a-1$. Cela implique aussi d'autres concepts temporels comme les intervalles de temps et les horloges.

Automates temporisés

Les modèles discrets peuvent être vus comme des automates. Dans ce type de structure, le temps est généralement intégré dans un formalisme appelé automates temporisés [75]. Dans ces automates, un état est décrit par deux vecteurs, un vecteur discret (correspondant à l'état dans l'automate discret) et un vecteur d'horloges continues (cf. Figure 4.5).

Pour modéliser les réseaux de régulation de gènes, une horloge h_u est associée à chaque gène u. Les horloges sont utilisées pour modifier la dynamique via des transitions gardées dans l'automate. Deux délais sont associés au couple $u_n = (u, n)$,

FIGURE 4.4 – Une fonction sigmoïdale (a) et ses approximations discrète (b) et linéaire par morceaux (c).

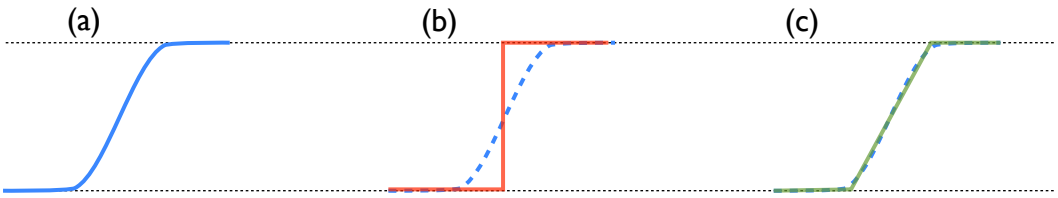
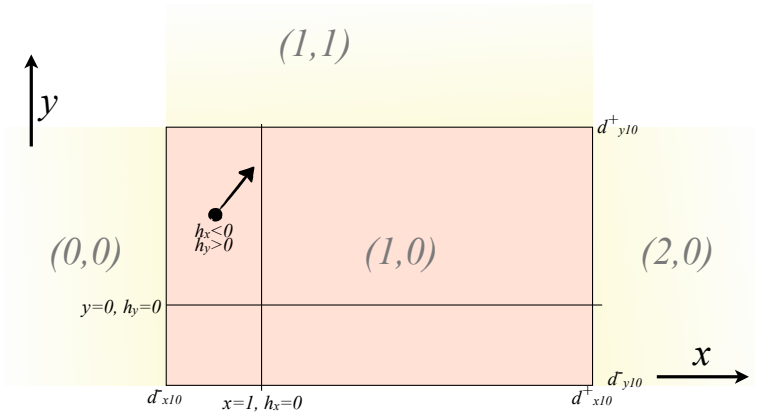


FIGURE 4.5 – Un exemple d'état défini par un automate hybride, ses régions temporelles et les délais. Ici, à l'état discret $(x, y) = (1, 0)$ sont associés les délais d^+_{x10}, d^-_{x10}, d^+_{y10} et d^-_{y10}.

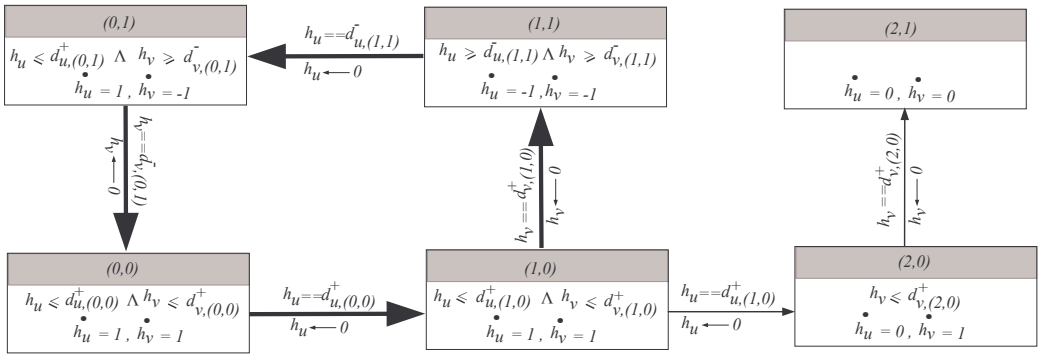


où u est un gène et n est son niveau d'expression ($d^+_{u_n}$ pour passer au niveau d'expression supérieur et $d^-_{u_n}$ pour passer au niveau inférieur). L'automate temporisé obtenu appartient à une classe d'automates temporisés appelée Linear Hybrid Automata (LHA) introduite par [76, 77]. La figure 4.6 en donne un exemple simple à deux gènes. C'est une extension temporisée du modèle donné en figure 4.2.

4.3.2 Vérification de modèle pour les automates temporisés

Ces modèles sont adaptés à une analyse automatique dont le but est de détecter les comportements cycliques ou identifier les *noyaux d'invariance*. Ces analyses peuvent être faite en utilisant un vérificateur de modèle. Dans cette étude, l'outil *HyTech* [78] a été utilisé. Son intérêt principal est de pouvoir travailler avec des paramètres (ici ce sont les délais, paramètres inconnus) en retournant les ensembles de contraintes que ces paramètres doivent vérifier pour assurer l'existence de certains comportements.

FIGURE 4.6 – Un automate temporisé correspondant à la production de mucus de *P. aeruginosa*. Son modèle discret est donné en figure 4.2.



Comportements

Nous donnons maintenant quelques définition plus formelles des caractéristiques de la dynamique temporelle que nous souhaitons extraire.

L'objet de base est la trajectoire. Une trajectoire est une séquence d'états reliés par des transitions à la fois discrètes et continues. Cette trajectoire peut-être associée à une fonction $\varphi : \mathbb{R}^+ \to \mathbb{S}$ où $\mathbb{S}$ est un pavé de dimension le nombre de gènes (*i.e.,*un pavé de $\mathbb{R}^n$) et dont la taille dépend des différents niveaux d'expression maximaux des gènes (par exemple, dans la figure 4.6, $\mathbb{S} = [0, 2] \times [0, 1]$). Intuitivement, l'automate hybride permet de calculer un champ de gradient pour $\mathbb{S}$, *i.e.,*à chaque point p de $\mathbb{S}$, un gradient (unique) est calculé via l'automate hybride qui permet de décrite l'évolution de ce point au cours d'un temps élémentaire dt.

Un cycle est une trajectoire qui commence et se termine par le même état hybride, *i.e.,*le même point de $\mathbb{S}$. Ici, c'est une extension importante des cycles discrets qui permet de travailler avec des spirales, convergentes ou divergentes.

On dit qu'une trajectoire est *viable* dans un ensemble $K \subset \mathbb{S}$ si et seulement si pour tout t, $\varphi(t) \in K$.

Attention, une trajectoire n'est pas nécessairement définie pour tout t. Il existe des trajectoires qui sortent de S, typiquement des spirales divergentes. Ces trajectoires ne sont donc pas viables dans S.

Soit $K \subset \mathbb{S}$, K est un *invariant* si pour tout point $p \in K$, la trajectoire qui commence en p reste dans K. Un *noyau d'invariance* K est un invariant maximal de $\mathbb{S}$ dans le sens où pour tout K' tel que K est inclu strictement dans K', K' n'est pas un invariant.

Le but des outils de vérification de modèle est de trouver ces cycles, noyaux d'invariance, et de fournir les conditions nécessaires à leur réalisation.

Quelques résultats chez *E. coli*.

Cette méthode a été appliqué au modèle d'Escherichia coli soumis à un stress de ressource carbonée. Plus précisément, dans [68], il a été montré que le modèle

discret exhibait deux phases, liées à deux cycles discrets, l'un correspondant à une phase de croissance exponentielle, l'autre à une phase d'équilibre.

Ces deux comportements ont pu être raffinés en ajoutant cette notion de délais. Voici un exemple de résultat qui peut être obtenu. Le cycle de la phase exponentielle n'est possible que si les 5 contraintes suivantes liant les délais des gènes fis et $gyrAB$ et la longueur totale L de ce cycle :

1. $d^+_{fis_1} + d^+_{fis_2} + d^+_{fis_3} + |d^-_{fis_3}| + |d^-_{fis_2}| \leq d^+_{gyr_{AB_0}} + d^+_{gyr_{AB_1}} + |d^-_{gyr_{AB_2}}|$
2. $d^+_{gyr_{AB_0}} + d^+_{gyr_{AB_0}} \leq d^+_{fis_1} + |d^-_{fis_2}| + |d^-_{fis_3}|$
3. $d^+_{gyr_{AB_0}} + d^+_{gyr_{AB_1}} + |d^-_{gyr_{AB_2}}| + |d^-_{gyr_{AB_1}}| \leq d^+_{fis_1} + d^+_{fis_2} + d^+_{fis_3} + |d^-_{fis_4}| + |d^-_{fis_3}| + |d^-_{fis_2}|$
4. $L = d^+_{fis_1} + d^+_{fis_2} + d^+_{fis_3} + |d^-_{fis_4}| + |d^-_{fis_3}| + |d^-_{fis_2}|$
5. $L = d^+_{gyr_{AB_0}} + d^+_{fis_1} + d^+_{fis_2} + d^+_{fis_3} + |d^-_{fis_4}|$.

4.3.3 Conclusions et perspectives

Dans ce travail, nous avons développé une méthodologie complète permettant de décrire très finement la dynamique d'un réseau de gène. La notion de délai qui a été introduite permet de se rapprocher des évolutions observées, et fournit des informations d'ordre quantitatif sur la dynamique qui doit être réellement observées.

Ensuite, à l'aide d'une approche de type vérification de modèle, des contraintes sur ces délais, a priori inconnus, sont obtenues. Elles sont associées à des comportements que le système doit satisfaire.

Une des questions qui reste alors ouverte est de donner une valeur fiable à ces paramètres. Sans avoir été exploitées dans ce sens, les méthodes que nous décrivons dans la suite de ce chapitre pourraient être appliquées pour obtenir des informations quantitatives sur les paramètres.

4.4 Event transition graph : impact d'une trajectoire sur l'environnement.

Un des travaux que nous menons actuellement part du constat suivant : les systèmes biologiques que nous souhaitons comprendre sont décrits à l'échelle d'un individu. C'est à cette échelle que le système doit être compris. Cependant, les observations qui permettent de valider ces modèles sont très souvent à une échelle totalement différente, l'échelle des populations. A cette échelle, des effets de lissage du caractère stochastique des individus doivent être pris en compte si l'on veut éviter les mauvaises interprétations. Prenons un exemple élémentaire : supposons que le système "ìndividuel" émette un signal parfaitement périodique (sinusoidal par exemple), sur une population d'individus désynchronisée, ce signal sera perçu (en moyenne) comme constant et donc observé comme tel.

De plus, les observations à disposition sont rarement exactement ce qu'il serait souhaitable d'observer en vue de valider le modèle (par exemple, les modifications de niveaux d'expression de tous les gènes du système au cours du temps) mais ce sont plutôt des traces produites par ces effets sur l'environnement (par exemple les modifications de concentrations de certaines protéines associées à ces gènes).

De ce constat est né la notion d'Event Transition Graph, une vision abstraite des évènements, tant génomiques que biochimiques, qui peuvent se produire et se succéder au niveau d'un individu. On suppose que même si on ne peut pas observer le déclenchement de tel ou tel autre évènement, chaque évènement laisse une trace sur l'environnement qui est observable.

Ledit graphe, obtenu de manière experte, tend à décrire toutes les successions de tels évènements qui peuvent se produire au niveau d'un individu. Tout le jeu consiste ensuite à déterminer la stratégie (qui peut-être qualifiée de stochastique) d'un individu, ce qui revient à déterminer les probabilités de transition entre les différents évènements. Pour déterminer ces dernières probabilités, il est fait usage de résultats en analyse en moyenne d'algorithmes qui permettent d'étudier les propriétés statistiques moyennes des "traces" laissées par les évènements (ces moyennes, représentatives de modifications au niveau de la population). Enfin, parce qu'il existe des relations implicites entre les probabilités de transition de l'individu et les valeurs moyennes des traces, il est possible d'obtenir des contraintes à satisfaire pour qu'un individu appartienne à une population dont on dispose de propriétés "moyennes". Dans la partie suivante, nous exprimons la philosophie de la démarche sur un exemple fictif. Ensuite nous donnons quelques résultats biologiques obtenus par cette méthode sur un modèle bactérien (*Escherichia coli* en réponse à un stress de privation de carbone).

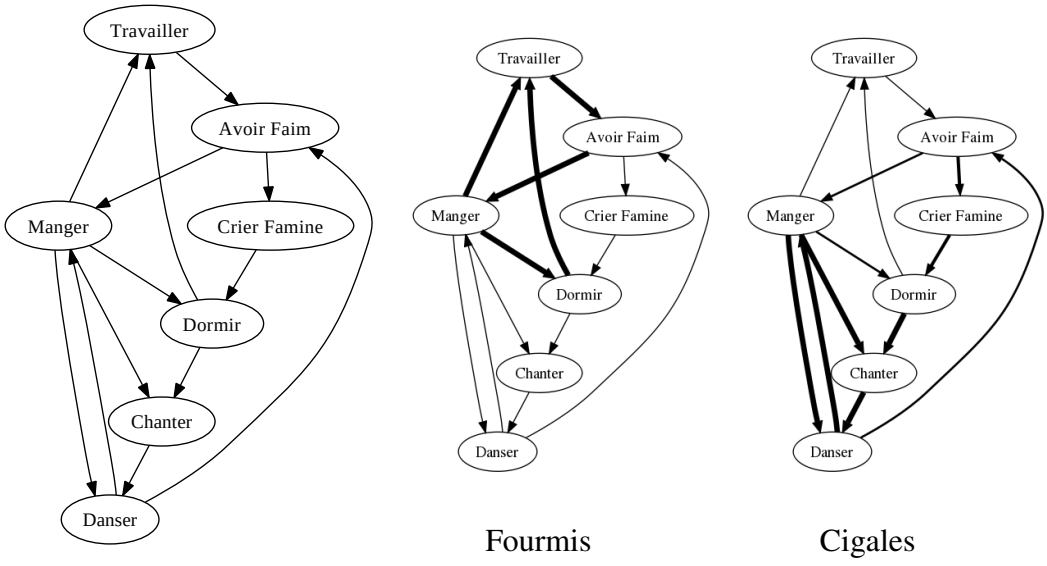
4.4.1 La cigale et la fourmi

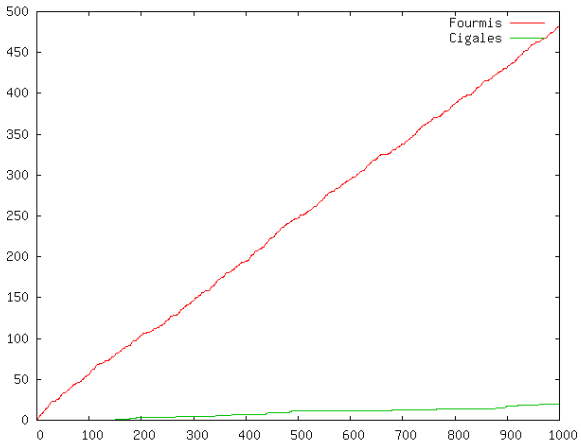
Nous allons maintenant illustrer la technique en s'appuyant sur un exemple très imagé inspiré de la fable de La Fontaine mais dont il ne faut pas chercher à tirer de conclusions importantes sur le modèle lui-même.

La figure 4.7 présente les actions et successions d'actions possibles d'une cigale et d'une fourmi.

La stratégie des fourmis (en tant que population homogène) pourra être décrite en ajoutant des probabilités de transition à toutes les successions d'évènements. On a évidemment affaire ici à une chaîne de Markov. Imaginons qu'il soit possible d'observer exactement toutes les actions d'une fourmi pendant un temps important, alors il est possible d'obtenir empiriquement les valeurs des probabilités de transition. Avec ces valeurs, on entre dans le cadre très simple des chaînes de Markov ergodiques (évidemment à condition que le squelette de la chaîne de Markov soit un graphe fortement connexe, c'est une hypothèse de la méthode mais comme dans le cas de la recherche de motifs, certaines extensions à des cas où cette contrainte est relachée sont envisageables). De nombreuses propriétés asymptotiques peuvent alors être dérivées. Ces propriétés peuvent concerner des trajectoires dans le graphe mais aussi des mesures plus quantitatives associées à ce graphe. Par exemple, considérons la quantité de réserve de nourriture. Cette quantité augmente en fonction des évènements qui se produisent (ou même des successions d'évènements). Par exemple, l'action de "Travailler" permet d'engranger 100g de nourriture, l'action de "Manger" en consomme 10g lorsque l'individu va se coucher (il s'agit d'un encas) mais elle en consomme 50g dans les autres cas. Ceci donne un modèle de coût dont il est facile de relier l'asymptotique avec les quantités propres de la chaîne de Mar-

FIGURE 4.7 – Le graphe des successions d'évènements commun et deux stratégies différentes. Puis les évolutions de la quantité des graines dans le grenier pour ces deux stratégies (exemples fictifs).





kov (comme le vecteur propre dominant ou la valeur propre dominante en suivant la méthode présentée dans [7]). La réponse apportée alors sera du type : "la quantité de nourriture en réserve de la fourmi augmente en moyenne de x grammes par jour". Toute l'idée de la méthode développée va être d'aborder ce même problème sous un angle légèrement différent. Et si maintenant les probabilités de transitions ne sont pas connues (en effet, en biologie, il est utopique de penser disposer de tels quantités). Par contre, il est facile d'observer l'évolution de la quantité de nourriture en réserve. Les questions deviennent donc :

- Si on fixe cette évolution, qu'elles peuvent être les probabilités de transition ? Finalement, ceci définit une contrainte numérique qui si elle peut être satisfaite mène généralement à un espace de solutions qu'il faut pouvoir explorer.
- Existe-t-il un jeu de probabilités de transition "optimal" ? Nous associerons

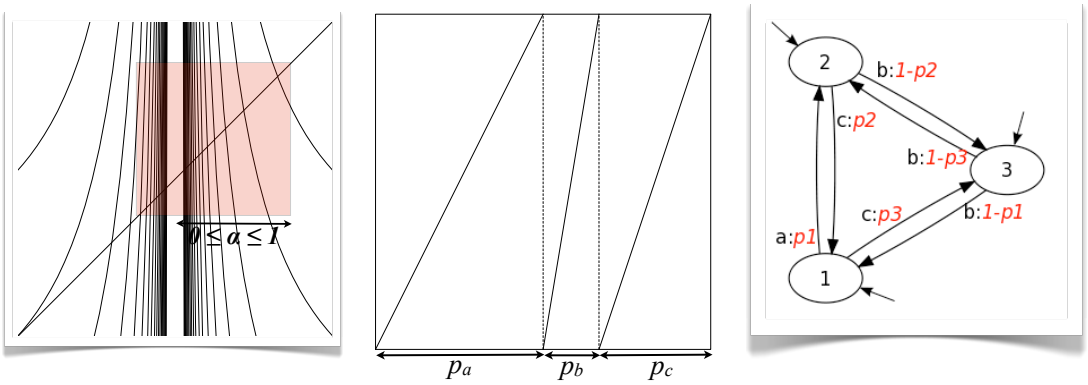
cette notion d'optimalité à une notion d'entropie maximale.
- Y a-t-il une manière permettant d'exprimer des coûts complexes ? Pour cela, nous utiliserons le vocabulaire des expressions régulières pour définir des coûts sur des langages réguliers dont les symboles sont les actions de la fourmi.

Pour résoudre ces questions à une grande échelle, nous avons développé des méthodes d'optimisation à base de recherche locale qui permettent de prendre en compte la définition implicite des contraintes.

4.4.2 Et les sources dynamiques dans tout ça

Tout le problème peut se reformuler dans le contexte des systèmes dynamiques. Le but ici est déterminer la meilleure source dynamique (celle de plus faible entropie), dont les trajectoires satisfont des propriétés asymptotiques fixées. Plus formellement, soit $\mathcal{S}$ une source dynamique. Cette source sera dite paramétrable si elle peut-être décrite par un ensemble fini de paramètres réels. Par exemple, une source sans mémoire est paramétrable, son ensemble de paramètres correspond à l'ensemble des probabilités des lettres. Une chaîne de Markov est également paramétrable, les paramètres qui permettent de la décrire sont bien évidemment les probabilités de transition de la chaîne de Markov ainsi que les probabilités initiales. Cette notion de source paramétrable s'étend à d'autres modèles de sources : une source dite "japonaise" [79] est paramétrable à l'aide d'un seul paramètre. Dans la suite, on note $\mathcal{S}(P)$ une source dynamique dont l'ensemble de paramètres est P. La figure 4.8 présente trois exemples de sources paramétrables. Dans [7], on a prouvé des relations entre la source dynamique (plus précisément des quantités liées à la source, des dérivées de valeurs propres dominantes par exemple) et des propriétés asymptotiques du nombre de positions d'occurrences d'un motif de type expression régulière parmi les mots aléatoires émis par la source. Lorsque la source est paramétrable, ces résultats s'expriment (de manière implicite car les valeurs propres dominantes sont définies comme telle) en fonction des paramètres P de la source. Il est à noter que lorsque tous les paramètres sont fixés, il est possible de calculer numériquement toutes les constantes qui interviennent dans les expressions asymptotiques. Cela est évident dans le cas des sources sans mémoire et des chaînes de Markov, le calcul est également possible dans le cas général en utilisant des techniques développées dans [43] ou encore [80]. Le théorème prouvé dans [7] s'étend de deux manières : (1) via une opération de multiplication d'automates, il est possible de combiner une source paramétrable et un automate qui porte des coûts élémentaires de transition pour obtenir une source paramétrable pondérée (le résultat de cette transformation, qui associe à une source paramétrable $\mathcal{S}(P)$ et un automate C, défini sur le même alphabet que la source et dont les transitions sont pondérées, est noté $\mathcal{S}_C(P)$) ; (2) de plus, ces coûts élémentaires de transition peuvent être combinés de deux manières, soit en les additionnant (comme dans le cas de la recherche de motifs), soit en les multipliant (ce qui permet de prendre en compte des augmentations en pourcentage, utiles en biologie notamment pour modéliser un effet dit de "protein burst effect").

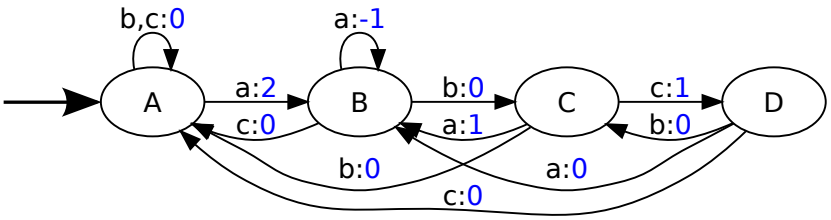
FIGURE 4.8 – Trois sources paramétrables : le système dynamique Japonais ($P = \{\alpha\}$), une source sans mémoire ($P = \{p_a, p_b, p_c\}$) ; et une chaîne de Markov ($P = \{p_1, p_2, p_3\}$)



Construction d'une source dynamique pondérée

Nous montrons ici comment obtenir une construction générale qui permet de passer d'un couple source dynamique et langage régulier pondéré à une source paramétrable pondérée (qui regroupe les informations de la source et des coûts). Ce langage régulier est donné par un automate pondéré qui décrit comment chaque mot du langage régulier doit être pondéré, la figure 4.9 donne un exemple d'automate pondéré. Cette construction est proche de celle qui se retrouve dans [7]. De la même manière que dans la section 3.2.2, on peut construire un opérateur matriciel $\mathbb{T}(u)$, tel que $\mathbb{T}_{i,j}(u) = u^{c_{i,j}} \sum_{m \in T_{i,j}} \mathbf{G}_{[m]}$, où $c_{i,j}$ est le coût de la transition pour passer de la transition i à la transition j dans l'automate pondéré et $T_{i,j}$ est l'ensemble des symboles qui permettent de passer de l'état i à l'état j dans cet automate.

FIGURE 4.9 – Un exemple d'automate pondéré. Ici, les mots `baabc`, `acabcb`, `bcaaa` ont comme poids respectifs 2, 5 et 0.

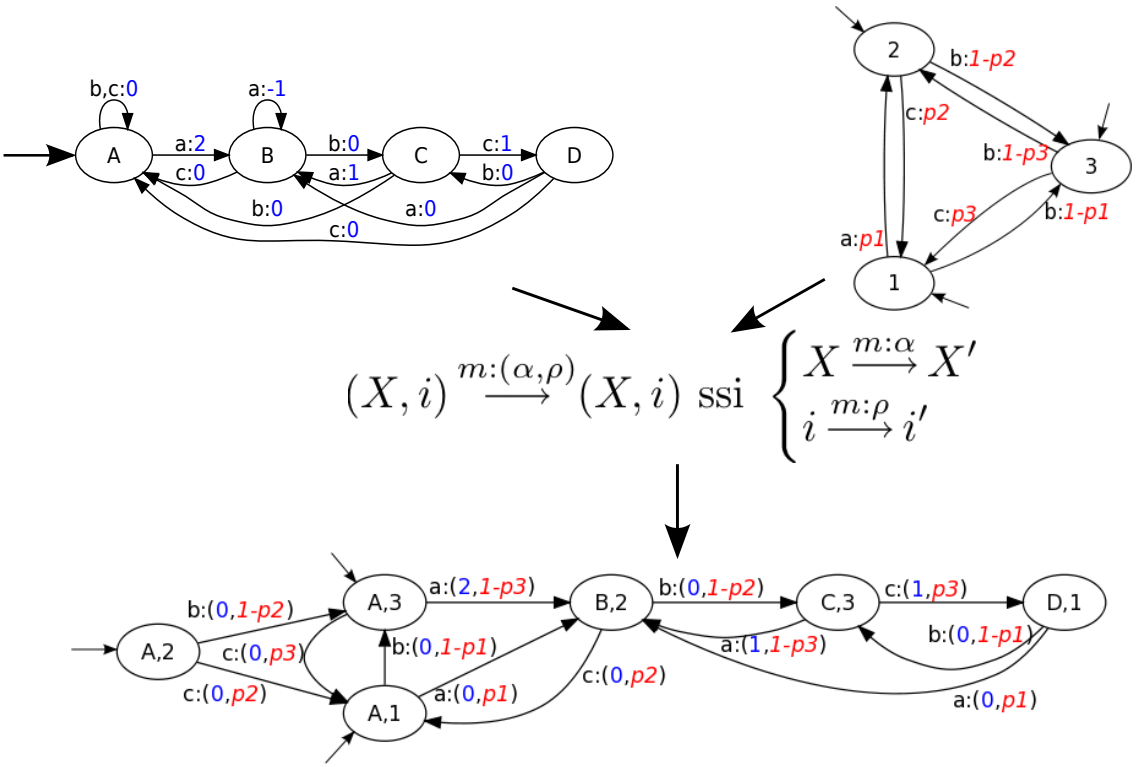


Lorsque la source est paramétrable, chaque opérateur $\mathbf{G}_{[m]}$ dépend (d'une partie) de l'ensemble des paramètres P. Lorsque la source est une chaîne de Markov, il est possible de pousser plus loin cette construction afin de faire apparaître plus explicitement les paramètres.

En effet, étant donnée une chaîne de Markov de matrice de transition M (dont certaines probabilités de transition sont inconnues, les éléments de l'ensemble de

paramètres P), on peut construire une chaîne de Markov pondérée de la manière suivante. Si on note $S = \{s_1, s_2, \ldots\}$ l'ensemble des états de la source et $A = \{a_1, a_2, \ldots\}$ l'ensemble des états de l'automate pondéré. Les états de la chaîne de Markov pondérée sont des éléments de $S \times A$ et il existe une transition entre l'état (s, a) et l'état (s', a') étiquetée par m, de coût c et de probabilité de transition p si et seulement s'il existe dans l'automate pondéré une transition de s vers s' de coût c et dans la chaîne de Markov une transition de a vers a' de probabilité p. Il est évident qu'avec cette construction, proche d'une composition de transducteurs, le résultat est effectivement une chaîne de Markov dans le sens où la somme des probabilités sortantes vaut 1. La figure 4.10 donne un exemple de telle construction.

FIGURE 4.10 – Construction d'une chaîne de Markov pondérée. Les probabilités initiales sont recopiées sur les états $(A, ?)$ car A est l'état initial de l'automate pondéré.



Désormais, nous disposons d'un objet mathématique qui permet d'étudier conjointement le coût d'un mot et sa probabilité. Nous allons maintenant voir comment utiliser cette structure.

Asymptotique des coûts dans une source paramétrable pondérée

Comme dans [7], une notion de "bonne" source apparaît. Dans le cas des sources paramétrables pondérées, ceci se traduit de la manière suivante : une source paramétrables pondérée est dite "bonne" si :

1. la source paramétrable qui est une source dynamique est bonne dans le sens de [7] ;
2. si l'automate C a pour support un graphe fortement connexe ;
3. s'il existe au moins deux cycles de même longueur et de coûts différents.

La troisième condition, nouvelle par rapport à [7] n'est là que pour assurer que le coût est bien une variable aléatoire. Elle est vérifiée de manière évidente dans le cas de la recherche de motifs.

Les bonnes sources paramétrables pondérées vérifient les théorèmes suivant, le premier valable pour les coûts additifs est adapté de [7], il se retrouve dans [16] :

Théorème 5 *Soit $\mathcal{S}_C(P)$ une "bonne" source paramétrable décomposable. Le coût additif d'un chemin de longueur n, noté C_n suit asymptotiquement une loi gaussienne quand $n \to +\infty$ dont le triplet caractéristique est donné par* $r[C_n] = O(1/\sqrt{n})$,

$$\mathrm{E}\,[C_n] = \gamma \cdot n + \gamma' + O(\mu^n) \quad \textit{et} \quad \mathbf{Var}(C_n) = \nu \cdot n + \nu' + O(\mu^n)$$

Les constantes γ et ν s'expriment en fonction de la pression $\Lambda(t)$ d'un operateur $\mathbb{T}$ qui dépend à la fois de la source et de l'automate, $\gamma = \Lambda'(0),\ \nu = \Lambda''(0)$, et $\mu < 1$ est n'importe quel réel strictement plus grand que le module de la valeur propre sous-dominante de $\mathbb{T}$.

La version multiplicative du théorème se prouve en utilisant les résultats sur les propriétés asymptotiques des graphes qui se trouvent dans [19].

Théorème 6 *Soit $\mathcal{S}_C(P)$ une "bonne" source paramétrable décomposable. Le coût additif d'un chemin de longueur n, noté C_n est tel que $\ln c_n$ suit asymptotiquement une loi gaussienne et il existe C tel que :*

$$\mathbb{E}(C_n^s) \approx C(e^s)\lambda^n(e^s) \textit{ pour } n \textit{ grand,}$$

où λ est la valeur propre dominante de $\mathbb{T}$ défini par $\mathbb{T}_{i,j} = \mathbb{G}_{i,j} * c(e_{i,j})$, où $\mathbb{G}_{i,j}$ est l'opérateur fonctionnel associé à la transition de i vers j et $c(e_{i,j})$ est le coût de la transition de i vers j. En particulier, on a :

$$\mathbb{E}(C_n) \approx C(e)\lambda^n(e) \textit{ pour } n \textit{ grand}$$
$$Var(C_n) \approx C(e^2)\lambda^n(e^2) - (C(e)\lambda^n(e))^2 \textit{ pour } n \textit{ grand.}$$

Dans la suite, nous utiliserons ces relations entre la moyenne d'un coût et les différentes valeurs propres des systèmes. Autrement dit, connaître la moyenne d'un coût donne des informations sur la valeur propre d'un système qui se transmettent aux paramètres de la source. Il faut tout de même noter que la relation entre valeur propre et probabilités de transition s'exprime de manière implicite.

4.4.3 Inférence de sources dynamiques paramétrables dont on connaît une pondération

Partant du principe qu'un phénomène biologique peut-être modélisé par une source dynamique, une question naturelle qui se pose est de déterminer la meilleure

source dynamique qui permet de représenter ce phénomène. C'est évidemment une question complexe qui implique d'une part de définir une mesure de qualité de la solution, d'autre part, de définir proprement un espace de recherche dans lequel se trouve cette meilleure source dynamique et enfin de mettre au point des méthodes de parcours de cet espace de solution adapté à sa topologie.

Expression du problème d'inférence

La première étape consiste à reformuler cette question sous la forme d'un problème d'optimisation numérique. Pour cela, il est nécessaire de se restreindre à la recherche à un ensemble de sources paramétrables par un ensemble de paramètres réels P. Dans la suite, on exprimera le problème de l'inférence pour des chaînes de Markov dont le squelette est donné mais dont les probabilités de transition sont inconnues. Le même travail peut être réalisé sans grande difficulté pour les sources dynamiques paramétrables. L'ensemble des paramètres sera donc l'ensemble $P = (p_{i,j})$ des probabilités de transition qui vérifient donc pour tout i, $\sum_j p_{i,j} = 1$ et $p_{i,j} = 0$ s'il n'existe pas de transition de i à j dans le squelette de la chaîne de Markov. Etant donné un coût, donné par son automate pondéré, nous avons vu dans 4.3.2.1 comment construire une chaîne de Markov pondérée et dans 4.3.2.2 comment les paramètres pouvaient être relié au coût asymptotique moyen. Il est possible d'obtenir une contrainte sur les paramètres en fixant le terme de premier ordre de l'asymptotique. La contrainte est du type

$$But = \gamma_P \text{ [cas additif] ou } But = \lambda_P(e) \text{ [cas multiplicatif].}$$

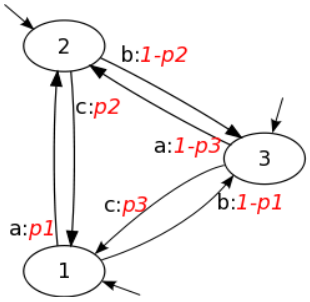
A chaque fois, on cherche à atteindre un certain taux de croissance (linéaire dans le cas additif ou exponentielle dans le cas multiplicatif). Les quantités γ_P et $\lambda_P(e)$, qui sont des objets propres dominants ont des expressions implicites en fonction de la matrice de transition de la chaîne de Markov, donc de P. Autrement dit, une des difficultés apportée par cette contrainte, c'est qu'il n'est pas possible d'obtenir, en général, d'expression analytique de ces contraintes. Cependant, il est possible de les calculer dès que les paramètres ont des valeurs fixées. Cette propriétés sera utilisée pour obtenir une méthode de résolution adaptée.

Dans la suite, on parlera de l'espace de solutions, c'est à dire l'ensemble des valeurs des paramètres qui vérifient les contraintes. C'est un sous-ensemble de l'hypercube unité de dimension $|P|$. La figure 4.11 donne un exemple simple de résolution.

Une question récurrente lorsque l'on considère les problèmes d'optimisation de ce type, pour lesquels il existe un ensemble continu de solutions est : "est-ce que parmi toutes ces solutions, il en existe une meilleure que les autres ?". En biologie, c'est une question d'ordre philosophique : est-ce qu'un système vivant possède un unique mode de fonctionnement "optimal" dans des conditions données ? Il est évidemment difficile de répondre à cette question mais imaginons que ce soit le cas et essayons de répondre à ce problème dans notre cas. La réponse apportée la plupart du temps est connue sous le nom de *Principe d'entropie maximale* (voir [81] pour une revue sur le sujet) qui consiste à privilégier, parmi tous les modèles aléatoires qui permettent de modéliser un problème, celle qui est d'entropie maximale. L'avantage de cette solution est que d'une part, elle introduit le minimum de connaissance

FIGURE 4.11 – Un exemple complet. Quelles sont les chaînes de Markov d'ordre 1 à trois lettres telles que les mots produits ne contiennent pas deux fois la même lettre consécutivement (contrainte sur le squelette) et telle que la proportion d'occurrences du motif cab est en moyenne égale à $1/6$.

Source



Matrice pondérée

(occurrences de cab)

$$M_p(u) = \begin{pmatrix} 0 & 0 & p_1 & 1-p_1 & 0 \\ p_2 & 0 & 0 & (1-p_2) & 0 \\ p_2 & 0 & 0 & 0 & (1-p_2)u \\ p_3 & 1-p_3 & 0 & 0 & 0 \\ p_3 & 1-p_3 & 0 & 0 & 0 \end{pmatrix}.$$

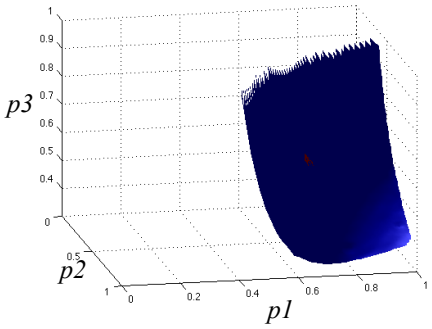
Contraintes

$0 < p_1, p_2, p_3 < 1$

$\lambda'_p(1) = 1/6$

où $\lambda_p(u)$ racine dominante de $\det(M_p(u) - X\ I) = 0$

Ensemble des pondérations valides

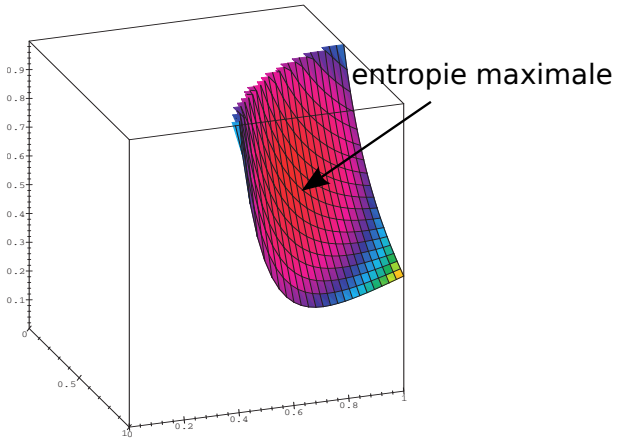


a priori dans la distribution, et d'autre part, il peut-être prouvé que cette solution est unique. Dans l'exemple de la figure 4.11, la solution d'entropie maximale est située au centre de l'ensemble de solutions (cf figure 4.12).

4.4.4 *Escherichia coli* en privation de carbone

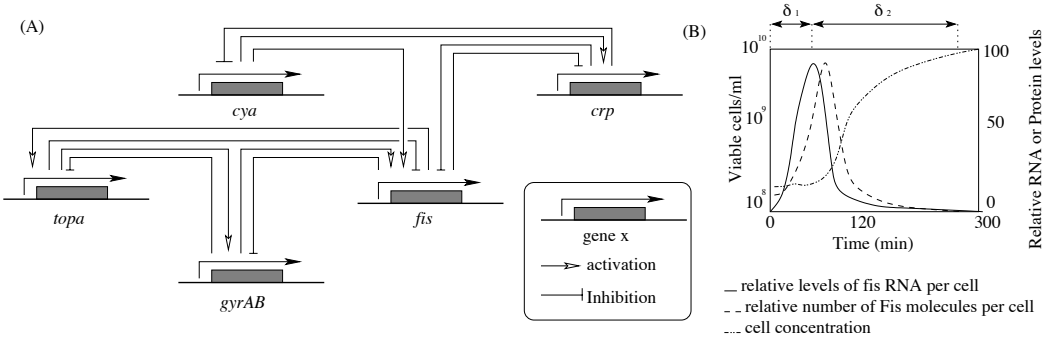
Nous appliquons la méthode exposée précédemment à un modèle classique de la biologie des systèmes. Ce modèle est tiré de [82] et les résultats présentés ici ont été publiés dans [15]. D. Ropers et ses collaborateurs ont modélisé la phase de croissance de la bactérie *Escherichia coli* après une période de stress nutritionel [82]. En particulier, leur modèle met en évidence deux phases (croissance exponentielle et phase stationnaire) et permet l'étude du passage entre les deux phases. Ce *"switch"* est mis en évidence au niveau de la régulation des gènes mais il à un impact au niveau du phenotype. Leur modèle est qualitatif, construit en utilisant des résultats bibliographiques et des expériences sur les régulations de gènes (il est décrit dans la figure 4.13). Il faut noter que d'autres expériences sont disponibles sur ce modèle. En effet, les protéines encodées par les gènes qui interagissent dans le réseau ont été bien étudiées dans des études indépendantes [83, 84]. Ces dernières études

FIGURE 4.12 – L'ensemble de solutions coloré en fonction de la valeur de l'entropie.



fournissent des informations quantitatives partielles qui pourraient être introduites dans le modèle qualitatif.

FIGURE 4.13 – Information biologique sur le système *Escherichia coli* en privation de carbone. (A) interactions entre les gènes du réseau (adapté de [82]). (B) Variations quantitatives de concentrations de molécules d'intérêt (basé sur [83]).

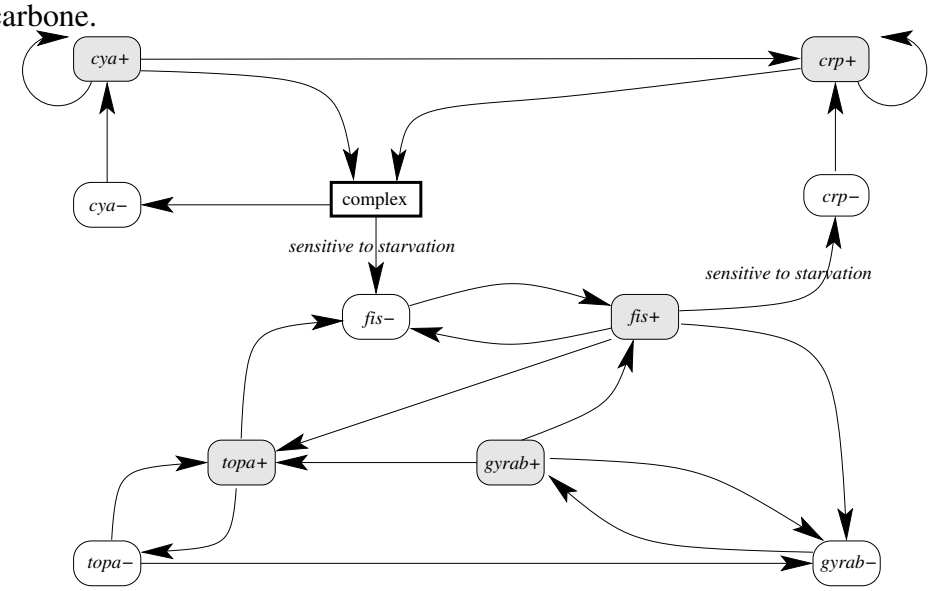


Event Transition Graph

Le modèle original de [82] est donné sous la forme d'un système d'équations différentielles linéaires par morceaux (PADE). Il contient 6 gènes et 37 contraintes sur des inégalités et des seuils. Il est possible, à partir de ce modèle de construire un Event Transition Graph automatiquement. Cette structure, dont la construction est décrite dans [15] permet d'abstraire la dynamique du système en considérant deux états possibles pour chaque gènes décrivant l'évolution de son activation (qui peut augmenter ou diminuer). Cette abstraction est un graphe qui décrit les successions possibles entre les évènements d'augmentation ou diminution de l'activité des gènes. Il est composé de 11 noeuds et 22 arêtes (cf. Figure 4.14). Il faut noter que ce dernier modèle a été "retouché" manuellement pour introduire un composant nomé

complexe qui résume l'effet d'un métabolite (cAMP) décrit dans [85]. Ce noeud est en accord avec [82] et correspond à une complexation des protéines Crp et Cya. Finalement, bien que cet ETG soit une abstraction très grossière du modèle original, il en préserve les propriétés biologiques principales. Par exemple, la répression du gène *crp* par la protéine Fis [86] est représentée par un effet actif de fis_+ sur $crp\_$. Ce modèle décrit clairement une chaîne de Markov de probabilités inconnues, donc une source dynamique paramétrable.

FIGURE 4.14 – ETG du modèle *E. coli*. Chaque composant représente un effet actif sur l'activation du gène. Deux transitions disparaissent lors d'un stress de privation de carbone.



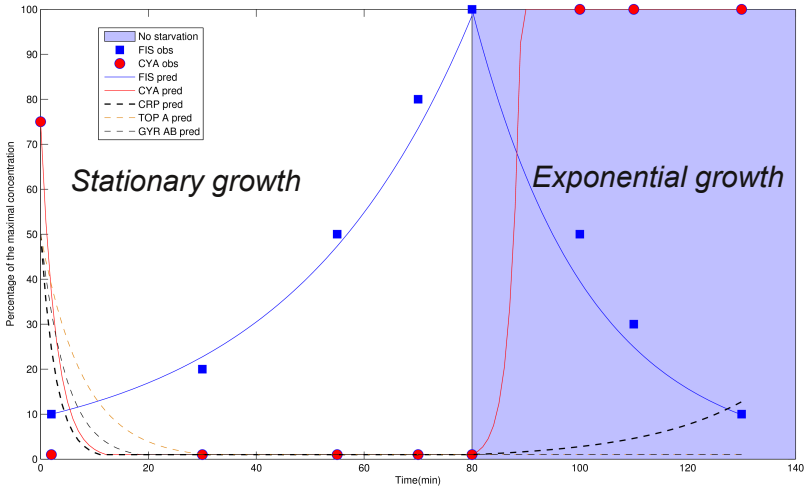
Pondérations

Ici, le système de pondération est très simple, pour chaque protéine, trois types de coûts sont intégrés et associés aux évènements eux mêmes : une augmentation, une diminution et une dégradation naturelle (diminution plus faible), tous les trois liés par une condition d'équilibre. Les coûts seront multiplicatifs, en accord avec le fait que pour les bactéries, il y a un effet d'amplification entre l'échelle transcriptomique et l'échelle protéique, phénomène appelé "protein burst effect" et décrit dans [87, 88, 89]. Dans notre exemple, pour la protéine Fis, le facteur d'augmentation est de 46% (à chaque fois qu'une transition menant à fis_+ est empruntée. Le facteur de diminution est de 32% pour toutes les transitions qui pointent vers $fis\_$, pour une dégradation naturelle de 5% attribuée à toutes les autres transitions.

Données à disposition et inférence

Pour inférer les paramètres de la chaîne de Markov, nous avons utilisé les données expérimentales de [83] qui montrent que la concentration de Fis est multipliée par 10 en 80 minutes (phase stationnaire) avant de décroitre, division de la quantité par 10 en 50 minutes. En utilisant uniquement ces trois points de mesure pour

FIGURE 4.15 – Simulation des changements des concentrations de protéines obtenues avec notre modèle dans les phases stationnaires et exponentielles. Après 80 minutes, le signal de privation de carbone est déclenché. Les données expérimentales sont dessinées en pointillés, les résultats des simulations sont tracés par des lignes pleines pour cinq protéines d'intérêt (Fis, Cya, Topa, GyrAB and Crp).



la protéine Fis, nous avons bien sûr retrouvé l'évolution observée pour Fis (validant la pondération et le caractère multiplicatif du coût) et nous avons obtenu une évolution très fidèle de la protéine Cya, observée dans [83]. Nous avons pu montrer que les protéines Cya et Crp ont le même comportement, ce que nous avons pu confirmer a posteriori par une étude bibliographique [90]. Enfin, en appliquant une notion de sensibilité de transition (basée sur les dérivées partielles des paramètres), nous avons pu extraire les transitions les plus sensibles (en gros, celles qui ont la plus forte dérivée partielle). En particulier, en phase stationnaire, la transition $fis_+ \rightarrow crp_-$ est très contrainte, ce qui est confirmé dans [86] car le complexe CAMP-CRP contrôle le métabolisme dans l'utilisation de sources de carbone alternative. Les transitions les plus sensibles sont données dans le tableau 4.1, avec à chaque fois une signification biologique de cette sensibilité.

TABLE 4.1 – Résumé des transitions les plus sensibles

Transition de l'ETG	Sensibilité	Signification biologique	Réf.
$fis_+ \rightarrow crp_-$	15.5%	control of CAMP-CRP complex	[86]
$gyrab_+ \rightarrow fis_+$	11.6%	*fis* regulation controlled by the DNA supercoiling level	[91]
$gyrab_+ \rightarrow topa_+$	8.1%	Topoisomerase I regulation by the DNA supercoiling	[92]
$fis_+ \rightarrow topa_+$	7.1%	Homeostatic control of DNA topology	[93, 94]
$fis_+ \rightarrow gyrab_-$	5.5%	Homeostatic control of DNA topology	[93, 94]
$gyrab_+ \rightarrow gyrab_-$	4.8%	*gyrAB* expression regulation by the DNA supercoiling	[93]

4.4.5 Conclusions et perspectives

Dans cette partie, nous avons montré qu'il était possible de combiner plusieurs échelles de description d'un système biologique en utilisant des pondérations de trajectoires dynamiques. Le transfert se fait en utilisant des théorèmes d'analyse en moyenne précis permettant de lier les caractéristiques de chaînes de Markov pondérées, un cas particulier de source dynamiques paramétrables pondérées, à des mesures sur des populations d'individus. Cette méthode a d'ors et déjà été exploitée pour étudier une exemple jouet, bien documenté et reconnu. Cependant, son intérêt est qu'elle ouvre des perpectives de recherche intéressantes dans de multiples directions. Par exemple, pour le moment, seuls des liens entre les coûts moyens sont utilisés pour définir des contraintes. Il serait aussi intéressant d'utiliser des temps t'atteinte par exemple (au bout de combien de temps une pondération dépasse-t-elle un certain seuil) ou encore de considérer des scores joints. Il reste également à résoudre des problèmes de passage à l'échelle de la méthode d'inférence qui est limitée actuellement à l'inférence de chaînes de Markov de moins de 30 états. Des propriétés de la pondération pourraient être utilisées pour en simplifier le calcul, d'une part, les algorithmes de recherche locale pourraient être affinés pour que la recherche soit plus efficace ou une architecture de calcul adaptée aux calculs matriciels lourd pourrait être utilisée, typiquement une architecture basée sur des cartes graphiques (unités GPU).

4.5 Vers une reconstruction (semi-)automatique de réseaux biologiques

Reconstruire automatiquement un réseau de gène, à partir de connaissances sur des séquences, de connaissances sur certaines interactions, etc,... est à l'heure actuelle un des challenges en biologie des systèmes. A tel point qu'une nouvelle thématique émerge actuellement dans ce domaine : on parle de *biologie intégrative* ou parfois de *génomique intégrative*. Les réseaux ainsi reconstruits, supposés fidèles aux mesures biologiques, auront des particularités fortes à prendre en compte dans toutes les études. Ils seront gros mais incomplets, souvent erronés et rarement utilisables directement par les outils "classiques" développés jusqu'à présent. Dans [95], nous proposons un cadre de reconstruction et d'analyse de réseaux métaboliques adapté à l'étude des algues brunes. Dans cette étude, qui prend la forme d'un survey, nous présentons notre point de vue concernant un cheminement permettant de distinguer pour une système vivant donné, ici l'algue brune *Ectocarpus Siliculosus*, les caractéristiques relevant d'une acclimatation à l'environnement (souvent d'ordre métabolique) de celles relevant d'une adaptation (plutôt d'ordre génétique). Nous avons exhibé des aspects méthodologiques qui doivent être résolus avant d'envisager de venir à bout d'une telle étude, comme par exemple : des problématiques de reconstruction de réseaux métabolique pour des organismes non modèles et des métabolismes peu connus (comme celui lié à la production de manitol, un substitut naturel à l'aspartame sécrété par ces algues et à fort potentiel industriel) ou encore l'analyse de réseaux métaboliques imparfaitement reconstruits (ici, impar-

fait signifie que la stœchiométrie n'est pas entièrement connues, des réactions sont manquantes ou erronées,...). Les méthodes classiques d'analyse doivent être adaptées à ces réseaux. C'est dans ce cadre que s'orienteront mes recherches futures en biologie des systèmes.

5

Conclusion générale et recherches futures

Dans tout ce manuscript, je me suis attaché à démontrer qu'il existait un lien très fort entre l'étude des séquences et la biologie des systèmes, ce qui est particulièrement vrai lorsque l'on s'intéresse à des propriétés de la dynamique des réseaux car dans ce cas, les objets les plus représentatifs sont des trajectoires suivies par le système. J'ai par exemple montré que des résultats obtenus sur les séquences peuvent être appliqués en biologie des systèmes afin de mieux en comprendre le fonctionnement. Ce transfert de résultats ouvre des perspectives importantes et je compte poursuivre dans cet axe.

Ainsi, mes recherches futures tendent à s'orienter vers une thématique bien identifiée : l'application de ***méthodes probabilistes pour l'étude des propriétés quantitatives des systèmes biologiques avec des applications aux stress environnementaux.***

En effet, mes travaux de ces dernières années et mon programme de recherche portent sur l'étude des systèmes biologiques. C'est un domaine où mes compétences en systèmes dynamiques, probabilités et statistiques sont importantes. Mon ambition est de définir des méthodes qui s'étendent de l'analyse théorique des réseaux d'interaction jusqu'aux applications à de vrais systèmes et questions biologiques. Pour cela, il est indispensable d'étudier les dynamiques des réseaux (pour avoir une vision qualitative des phénotypes associés aux systèmes) ; de définir des modèles quantitatifs (stochastiques pour prendre en compte des données biologiques très bruitées ou incomplètes) ; et définir des méthodes automatiques de construction de réseaux biologiques. Tout ceci devant être réalisé sans s'éloigner des problématiques biologiques, et donc en établissant un dialogue constant entre modélisateurs et expérimentateurs. De manière concrète, je compte d'une part m'employer à développer des méthodes probabilistes pour l'étude qualitative et quantitative de la dynamique discrète des réseaux de régulation de gènes. D'autre part, j'ai entamé dernièrement des collaborations fructueuses autour de l'étude de réseaux métaboliques

et/ou à base de règles biochimiques, avec des applications en nutrition animale (des résultats préliminaires sont parus [96], d'autres sont en cours de soumission [97]).

Ainsi, mon programme de recherche s'inscrit à long terme dans le développement et l'application concrète de techniques probabilistes théoriques pour l'analyse de la dynamique de réseaux biologiques discrets. Il s'articule à court et moyen terme autour de questions de mise en commun des différentes méthodes développées dans les deux branches. Je compte aborder ces quatre axes :

1. *Extraction d'informations.* Une des étapes indispensable à toute modélisation concerne l'extraction d'informations pertinentes à partir de données hétérogènes. Les données à traiter actuellement sont produites à haut-débit et sont très bruitées. Il est indispensable d'adapter les méthodes existantes en prenant en compte ces caractéristiques. Il faut définir des méthodes probabilistes et statistiques précises. C'est une thématique que je compte développer via des collaborations tant avec des équipes de biologistes qu'avec des équipes de bioinformatique, spécialistes des calculs probabilistes et de l'extraction de connaissance. C'est une problématique complémentaire des études de modélisation dynamique puisque son but est d'extraire des informations complexes à partir de jeux de données hétérogènes, débouchant sur des propriétés que les modèles doivent vérifier.
2. *Modélisation.* Je pense qu'il est maintenant possible d'établir des ponts entre les deux types de modélisation formelles et probabilistes. Une solution pour y parvenir va consister à étendre le cadre des modèles probabilistes (discret et plus seulement booléen ; intégrant des notions complexes de synchronisation ; prenant en compte des délais de réaction). Il s'agit aussi d'adapter tant les méthodes d'inférence que les méthodes de vérification à ce nouveau modèle. Les algorithmes d'inférence utilisés ici sont fortement inspirés de ceux utilisés lors de l'inférence d'automates. Le challenge principal va également consister à obtenir une interprétation fine des notions de synchronisation, telles que celles définies dans ce manuscript.
3. *Réduction de modèle.* Quel que soit le modèle envisagé, l'objet final d'étude est le graphe d'état qui est de taille exponentielle en le nombre de gènes. C'est actuellement un des verrous à l'étude de gros modèles. Je compte développer des méthodes de réduction de modèle d'une part, et appliquer des méthodes algorithmiques adaptées pour les traiter. Par exemple, beaucoup d'opérations sont basées sur des multiplications de grosses matrices, je suis convaincu qu'il est nécessaire d'utiliser des architectures hautement parallèles comme les cartes graphiques de nouvelle génération. Je suis actuellement en train d'adapter une méthode d'inférence de modèle (méthode que nous avons appelée POGG) à ce type d'architecture.
4. *Multi-échelle.* Regrouper toutes les échelles du vivant dans un même modèle est actuellement un des Graal de la biologie des systèmes. Dernièrement, nous avons développé une méthode très flexible de pondération d'évènements pour modéliser les modifications protéiques à l'échelle du gène. Grossièrement, ceci est réalisé en ajoutant des pondérations à chaque modification de l'activité des gènes, puis en étudiant les propriétés asymptotiques de ces pondérations par des méthodes probabilistes. Les résultats préliminaires obtenus

sur les bactéries sont très prometteurs mais des efforts de formalisation et d'analyse (pour extraire d'autres propriétés probabilistes) sont nécessaires. Je compte poursuivre cette étude, notamment via un co-encadremment de thèse débuté en septembre 2012.

Les deux dernières thématiques sont très concurrentielles, j'y apporte une vision différente, tant méthodologique (utilisation de techniques d'analyse en moyenne) qu'algorithmique, qui est à mon avis manquante dans les études actuelles.

Enfin, les demandes et attentes des biologistes sont actuellement gigantesques. Je compte placer une partie de mes efforts sur le transfert de méthodes purement théoriques vers des *applications biologiques en écologie et en génomique.*

Bibliographie

[1] DURBIN (R.), EDDY (S.), KROGH (A.) et MITCHISON (G.), *Biological sequence analysis*. Press, Cambridge U., édition eleventh, 2006. 4

[2] DANDEKAR (T.), SNEL (B.), HUYNEN (M.) et BORK (P.), « Conservation of gene order : a fingerprint of proteins that physically interact », *Trends Biochem. Sci.*, vol. 23, n° 9, Sep 1998, p. 324–328. 5

[3] ENRIGHT (A. J.), ILIOPOULOS (I.), KYRPIDES (N. C.) et OUZOUNIS (C. A.), « Protein interaction maps for complete genomes based on gene fusion events », *Nature*, vol. 402, n° 6757, Nov 1999, p. 86–90. 5

[4] WAGNER (A.), « Genes regulated cooperatively by one or more transcription factors and their identification in whole eukaryotic genomes », *Bioinformatics*, vol. 15, n° 10, Oct 1999, p. 776–784. 5

[5] MITROPHANOV (A. Y.) et BORODOVSKY (M.), « Statistical significance in biological sequence analysis », *Briefings in Bioinformatics*, vol. 7, n° 1, 2006, p. 2–24. 5

[6] BOURDON (J.) et MANCHERON (A.), « Statistical Properties of Similarity Score Functions », dans *Proceedings of the 4th Colloquium on Mathematics and Computer Science. Algorithms, Trees, Combinatorics and Probabilities, Discrete Mathematics and Theoretical Computer Science*, coll. « DMTCS Proceedings », p. 129–140, France, septembre 2006. 5, 28

[7] BOURDON (J.) et VALLÉE (B.), « Pattern matching statistics on correlated sources », dans *Proceedings of LATIN 2006*, vol. 3887, p. 224–237, 2006. 5, 39, 40, 62, 63, 64, 65, 66

[8] BOURDON (J.) et VALLÉE (B.), « Generalized pattern matching statistics », dans BIRKHAUSER (T. i. M.), éditeur, *Mathematics and Computer Science II*, p. 249–265, 2002. 5, 37, 38, 39

[9] BOURDON (J.) et RUSU (I.), « Statistical properties of factor oracles », *J. Discrete Algorithms*, vol. 9, n° 1, 2011, p. 57–66. 5

[10] KIM (S.), LI (H.), DOUGHERTY (E. R.) *et al.*, « Can Markov Chain Models Mimic Biological Regulation ? », *J Biol. Syst.*, vol. 10, n° 4, 2003, p. 337–357. 7

[11] SHMULEVICH (I.), GLUHOVSKY (I.), HASHIMOTO (R. F.) *et al.*, « Steady-state analysis of genetic regulatory networks modelled by probabilistic boolean networks. », *Comp. Funct. Genom.*, vol. 4, n° 6, 2003, p. 601–608. 7

[12] KAUFFMAN (S. A.), « Metabolic stability and epigenesis in randomly constructed genetic nets », *J. Theor. Biol.*, vol. 22, Mar 1969, p. 437–467. 7

[13] MERLE (T.) et BOURDON (J.), « Complex update strategies for Probabilistic Boolean Networks », dans TISCP, éditeur, *7th Workshop on Computational Systems Biology*, p. 4, Luxembourg, France, juin 2010. 7

[14] AHMAD (J.), BOURDON (J.), EVEILLARD (D.) *et al.*, « Temporal constraints of a gene regulatory network : Refining a qualitative simulation », *Biosystems*, vol. 98, n° 3, 2009, p. 149–159. 8

[15] BOURDON (J.), EVEILLARD (D.) et SIEGEL (A.), « Integrating quantitative knowledge into a qualitative gene regulatory network », *PLoS Computational Biology*, vol. 7, n° 9, 2011. 8, 68, 69

[16] BOURDON (J.) et EVEILLARD (D.), « Toll based measures for dynamical graphs ». Rapport technique n° q-bio/0702060, arXiv, 2007. 8, 66

[17] SZPANKOWSKI (W.), *Average Case Analysis of Algorithms on Sequences*. John Wiley & Sons, Inc., New York, NY, USA, 2001. 11

[18] LOTHAIRE (M.), *Applied Combinatorics on Words*. Cambridge University Press, 2005. 11, 39

[19] FLAJOLET (P.) et SEDGEWICK (R.), *Analytic Combinatorics*. Cambridge University Press, 2009. 11, 24, 66

[20] CLAUDE (S.), « Prediction and entropy of printed english », *Bell System Technical Journal*, vol. 30, 1951, p. 50–64. 15

[21] WELSH (D.), *Codes and cryptography*. The Clarendon Press Oxford University Press, New York, 1988. 15

[22] VALLÉE (B.), « Dynamical sources in information theory : fundamental intervals and word prefixes », *Algorithmica*, vol. 29, n° 1-2, 2001, p. 262–306. 15, 21

[23] FLAJOLET (P.) et VALLÉE (B.), « Continued fraction algorithms, functional operators, and structure constants », *Theoret. Comput. Sci.*, vol. 194, n° 1-2, 1998, p. 1–34. 21

[24] FLAJOLET (P.) et VALLÉE (B.), « Continued fractions, comparison algorithms, and fine structure constants », dans *Constructive, experimental, and nonlinear analysis (Limoges, 1999)*, vol. 27 (coll. *CMS Conf. Proc.*), p. 53–82. Amer. Math. Soc., Providence, RI, 2000. 21

[25] BOURDON (J.) et VALLÉE (B.), « Generalized pattern matching statistics », dans BIRKHAUSER (T. i. M.), éditeur, *Mathematics and Computer Science II*, p. 249–265, 2002. 21

[26] HWANG (H.-K.), *Théorèmes limites pour les structures combinatoires et les fonctions arithmétiques*. Thèse de doctorat, Ecole Polytechnique, Palaiseau, France, 1994. 24

[27] RIGOUTSOS (I.), FLORATOS (A.), PARIDA (L.) *et al.*, « The emergence of pattern discovery techniques in computational biology », *Journal of Metabolic Engineering*, vol. 2, n° 3, 2000, p. 159–177. 39

[28] VANET (A.), MARSAN (L.) et SAGOT (M.-F.), « Promoter sequences and algorithmical methods for identifying them », *Research in Microbiology*, vol. 150, 1999, p. 779–799. 39

[29] WATERMAN (M. S.), *Introduction to Computational Biology*. Chapman & Hall, 1995. 39

[30] NICODÈME (P.), DOERKS (T.) et VINGRON (M.), « Proteome analysis based on motif statistics », *Bioinformatics*, vol. 18, 2002, p. 161–171. Presented at the First European Conference on Computational Biology ECCB02, Saarbrücken, October 2002. 39

[31] GUIBAS (L. J.) et ODLYZKO (A. M.), « Periods in strings », *J. Combin. Theory Ser. A*, vol. 30, n° 1, 1981, p. 19–42. 39

[32] GUIBAS (L. J.) et ODLYZKO (A. M.), « String overlaps, pattern matching, and nontransitive games », *J. Combin. Theory Ser. A*, vol. 30, n° 2, 1981, p. 183–208. 39

[33] RÉGNIER (M.) et SZPANKOWSKI (W.), « On the approximate pattern occurrences in a text », dans Proc. Compression and Complexity of SEQUENCE'97, IEEE Computer Society, p. 253–264, 1997. 39

[34] RÉGNIER (M.) et SZPANKOWSKI (W.), « On pattern frequency occurrences in a Markovian sequence », *Algorithmica*, vol. 22, n° 4, 1998, p. 631–649. 39

[35] BASSINO (F.), CLÉMENT (J.), FAYOLLE (J.) et NICODÈME (P.), « Constructions for Clumps Statistics. », dans *Proceedings of the 5th International Colloquium on Mathematics and Computer Science (MathInfo'08)*, vol. AG (coll. *Discrete Mathematics and Theoretical Computer Science Proceedings*), p. 183–198., Blaubeuren, Allemagne, septembre 2008. DMTCS. 39

[36] SCHBATH (S.), « An overview on the distribution of word counts in markov chains », *Journal of Computational Biology*, vol. 7, n° 1-2, 2000, p. 193–201. 39

[37] ROBIN (S.), SCHBATH (S.) et VANDEWALLE (V.), « Statistical tests to compare motif count exceptionalities », *BMC Bioinformatics*, vol. 8, 2007. 39

[38] BOEVA (V.), CLÉMENT (J.), RÉGNIER (M.) et VANDENBOGAERT (M.), « Assessing the significance of sets of words », dans APOSTOLICO (A.), CROCHEMORE (M.) et PARK (K.), éditeurs, *CPM*, vol. 3537 (coll. *Lecture Notes in Computer Science*), p. 358–370. Springer, 2005. 39

[39] FLAJOLET (P.), SZPANKOWSKI (W.) et VALLÉE (B.), « Hidden word statistics », *to appear in Journal de l'ACM*, 2005. 39

[40] NICODÈME (P.), SALVY (B.) et FLAJOLET (P.), « Motif statistics », *Theoretical Computer Science*, vol. 287, n° 2, 2002, p. 593–617. 39

[41] NUEL (G.), « S-spatt : simple statistics for patterns on markov chains », *Bioinformatics*, vol. 21, n° 13, 2005, p. 3051–3052. 39

[42] NUEL (G.), « Effective p-value computations using finite markov chain imbedding (fmci) : application to local score and to pattern statistics », *Algorithms for Molecular Biology*, vol. 1, 2006. 39

[43] BOURDON (J.), *Analyse dynamique d'algorithmes : exemples en arithmétique et en théorie de l'information*. Thèse de doctorat, Université de Caen, 2002. 39, 63

[44] ALLAUZEN (C.), CROCHEMORE (M.) et RAFFINOT (M.), « Factor oracle : A new structure for pattern matching », dans *SOFSEM*, p. 295–310, 1999. 42, 43

[45] JACQUET (P.) et SZPANKOWSKI (W.), « Analytical depoissonization and its applications. », *Theor. Comput. Sci.*, vol. 201, n° 1-2, 1998, p. 1–62. 45

[46] DE JONG (H.), « Modeling and simulation of genetic regulatory systems : a literature review », *Journal of Computational Biology*, vol. 9, n° 1, 2000, p. 67–103. 48

[47] ENDY (D.) et BRENT (R.), « Modelling cellular behavior », *Nature*, vol. 409, 2001, p. 391–395. 48

[48] HASTY (J.), MCMILLEN (D.), ISAACS (F.) et COLLINS (J.), « Computational studies of gene regulatory networks », *Nat. Rev. Genet.*, vol. 2, 2001, p. 268–279. 48

[49] SMOLEN (P.), BAXTER (D.) et J.H. (B.), « Modeling transcriptional control in gene networks : Methods, recent results and future directions », *Bull. Math. Biol.*, vol. 62, 2000, p. 247–292. 48

[50] MCADAMS (H.) et ARKIN (A.), « Simulation of prokaryotic genetic circuits », *Ann. Rev. Biophys. Biomol. Struct.*, vol. 27, 1998, p. 199–224. 48

[51] DRULHE (S.), FERRARI-TRECATE (G.), DE JONG (H.) et VIARI (A.), « Reconstruction of switching thresholds in piecewise-affine models of genetic regulatory networks », dans HESPANHA (J. P.) et TIWARI (A.), éditeurs, *HSCC*, vol. 3927 (coll. *Lecture Notes in Computer Science*), p. 184–199. Springer, 2006. 49

[52] HUANG (S.), « Gene expression profiling, genetic networks, and cellular states : An integrating concept for tumorigenesis and drug discovery », *J. Mol. Med.*, vol. 77, 1999, p. 469–480. 49

[53] SOMOGYI (R.) et SNIEGOSKI (C.), « Modeling the complexity of genetic networks : Understanding multigenic and pleiotropic regulation », *Complexity*, vol. 1, n° 6, 1996, p. 45–63. 49

[54] KAUFMANN (S.), *The origins of Order : Self-Organization and Selection in Evolution*. Oxford University Press, New York, 1993. 49

[55] KAUFFMAN (S.), « Metabolic stability and epigenesis in randomly constructed genetic nets », *J Theor Biol*, vol. 22, n° 3, Mar 1969, p. 437–67. 49, 50

[56] THOMAS (R.), THIEFFRY (D.) et KAUFMAN (M.), « Dynamical behaviour of biological regulatory networks : I. biological role of feedback loops and practical use of the concept of the loop-characteristic state », *Bull. Math. Biol.*, vol. 57, n° 2, 1995, p. 247–276. 49

[57] SHMULEVICH (I.), DOUGHERTY (E. R.), KIM (S.) et ZHANG (W.), « Probabilistic boolean networks : a rule-based uncertainty model for gene regulatory networks », *Bioinformatics*, vol. 18, n° 2, Feb 2002, p. 261–74. 49

[58] SHMULEVICH (I.) et ZHANG (W.), « Binary analysis and optimization-based normalization of gene expression data », *Bioinformatics*, vol. 18, n° 4, Apr 2002, p. 555–65. 49

[59] SHMULEVICH (I.), DOUGHERTY (E.), KIM (S.) et ZHANG (W.), « Probabilistic boolean networks : a rule-based uncertainty model for gene regulatory networks », *Bioinformatics*, vol. 18, n° 2, Feb 2002, p. 261–74. 49, 51

[60] GLASS (K.) et KAUFFMAN (S.), « The logical analysis of continuous, nonlinear biochemical control networks », *J. Theor. Biol.*, vol. 39, 1973, p. 103–129. 50

[61] HUANG (S.), « Gene expression profiling, genetic networks, and cellular states : an integrating concept for tumorigenesis and drug discovery », *J Mol Med*, vol. 77, n° 6, Jun 1999, p. 469–80. 51

[62] HUANG (S.), « Genomics, complexity and drug discovery : insights from boolean network models of cellular regulation », *Pharmacogenomics*, vol. 2, n° 3, Aug 2001, p. 203–22. 51

[63] DENG (X.), GENG (H.) et MATACHE (M. T.), « Dynamics of asynchronous random Boolean networks with asynchrony generated by stochastic processes », *BioSystems*, vol. 88, Mar 2007, p. 16–34. 52

[64] THOMAS (R.), THIEFFRY (D.) et KAUFMAN (M.), « Dynamical behaviour of biological regulatory networks–i. biological role of feedback loops and practical use of the concept of the loop-characteristic state », *Bull Math Biol*, vol. 57, n° 2, Mar 1995, p. 247–76. 52, 55

[65] THIEFFRY (D.) et THOMAS (R.), « Dynamical behaviour of biological regulatory networks–ii. immunity control in bacteriophage lambda », *Bull Math Biol*, vol. 57, n° 2, Mar 1995, p. 277–97. 52, 55

[66] NALDI (A.), BERENGUIER (D.), FAURÉ (A.) *et al.*, « Logical modelling of regulatory networks with ginsim 2.3 », *Biosystems*, May 2009. 53

[67] AHMAD (J.), BOURDON (J.), EVEILLARD (D.) *et al.*, « Temporal constraints of a gene regulatory network : Refining a qualitative simulation », *BioSystems*, vol. 98, n° 3, Dec 2009, p. 149–159. 56

[68] ROPERS (D.), DE JONG (H.), PAGE (M.) *et al.*, « Qualitative simulation of the carbon starvation response in escherichia coli », *Biosystems*, vol. 84, n° 2, May 2006, p. 124–52. 57, 59

[69] BERNOT (G.), COMET (J. P.), RICHARD (A.) et GUESPIN (J.), « Application of formal methods to biological regulatory networks : extending Thomas' asynchronous logical approach with temporal logic », *J. Theor. Biol.*, vol. 229, n° 3, Aug 2004, p. 339–347. 57

[70] ADÉLAÏDE (M.) et SUTRE (G.), « Parametric analysis and abstraction of genetic regulatory networks », dans *Proc. 2nd Workshop on Concurrent Models in Molecular Biology (BioCONCUR'04), London, UK, Aug. 2004*, coll. « Electronic Notes in Theor. Comp. Sci. ». Elsevier, 2004. 57

[71] SIEBERT (H.) et BOCKMAYR (A.), « Incorporating time delays into the logical analysis of gene regulatory networks », dans PRIAMI (C.), éditeur, *CMSB*, vol.

4210 (coll. *Lecture Notes in Computer Science*), p. 169–183. Springer, 2006. 57

[72] KLARNER (H.), SIEBERT (H.) et BOCKMAYR (A.), « Time Series Dependent Analysis of Unparametrized Thomas Networks », *IEEE/ACM Trans Comput Biol Bioinform*, Apr 2012. 57

[73] AHMAD (J.), ROUX (O.), BERNOT (G.) *et al.*, « Analysing formal models of genetic regulatory networks with delays », *International Journal of Bioinformatics Research and Applications (IJBRA)*, vol. 4, n° 2, 2008. 57

[74] AHMAD (J.), BERNOT (G.), COMET (J.-P.) *et al.*, « Hybrid modelling and dynamical analysis of gene regulatory networks with delays », *ComPlexUs*, vol. 3, n° 4, octobre 2007, p. 231–251. 57

[75] ALUR (R.) et DILL (D.), « A theory of timed automata », *Theoretical Computer Science*, vol. 126, 1994, p. 183–235. 57

[76] HENZINGER (T.) et HO (P.-H.), « Algorithmic analysis of nonlinear hybrid systems », dans *CAV : Computer-Aided Verification*, coll. « Lecture Notes in Computer Science 939 », p. 225–238. Springer, 1995. 58

[77] HENZINGER (T.), « The theory of hybrid automata », dans *Proceedings of the 11th Annual Symposium on Logic in Computer Science*, p. 278–292. IEEE Computer Society Press, 1996. 58

[78] HENZINGER (T.-A.), HO (P.-H.) et WONG-TOI (H.), « HYTECH : A model checker for hybrid systems », *International Journal on Software Tools for Technology Transfer*, vol. 1, n° 1–2, 1997, p. 110–122. 58

[79] BOURDON (J.), DAIREAUX (B.) et VALLÉE (B.), « Dynamical analysis of alpha-euclidean algorithms », *J. Algorithms*, vol. 44, n° 1, 2002, p. 246–285. 63

[80] LHOTE (L.), « Computation of a class of continued fraction constants », dans ARGE (L.), ITALIANO (G. F.) et SEDGEWICK (R.), éditeurs, *ALENEX/ANALC*, p. 199–210. SIAM, 2004. 63

[81] HARREMOËS (P.) et TOPSØE (F.), « Maximum entropy fundamentals », *Entropy*, vol. 3, n° 3, 2001, p. 191–226. 67

[82] ROPERS (D.), DE JONG (H.), PAGE (M.) *et al.*, « Qualitative simulation of the carbon starvation response in Escherichia coli », *BioSystems*, vol. 84, n° 2, May 2006, p. 124–52. 68, 69, 70

[83] BALL (C. A.), OSUNA (R.), FERGUSON (K. C.) et JOHNSON (R. C.), « Dramatic changes in Fis levels upon nutrient upshift in *Escherichia coli* », *J Bacteriol*, vol. 174, n° 24, Dec 1992, p. 8043–56. 68, 69, 70, 71

[84] NOTLEY-MCROBB (L.), DEATH (A.) et FERENCI (T.), « The relationship between external glucose concentration and cAMP levels inside *Escherichia coli* : implications for models of phosphotransferase-mediated regulation of adenylate cyclase », *Microbiology+*, vol. 143, 1997, p. 1909–1918. 68

[85] HARMAN (J. G.), « Allosteric regulation of the cAMP receptor protein », *Biochim. Biophys. Acta*, vol. 1547, n° 1, May 2001, p. 1–17. 70

[86] GONZÁLEZ-GIL (G.), KAHMANN (R.) et MUSKHELISHVILI (G.), « Regulation of crp transcription by oscillation between distinct nucleoprotein complexes », *EMBO J.*, vol. 17, n° 10, May 1998, p. 2877–85. 70, 71

[87] CAI (L.), FRIEDMAN (N.) et XIE (X. S.), « Stochastic protein expression in individual cells at the single molecule level. », *Nature*, vol. 440, n° 7082, mar 2006, p. 358–362. 70

[88] TANIGUCHI (Y.), CHOI (P. J.), LI (G.-W.) *et al.*, « Quantifying *E. coli* proteome and transcriptome with single-molecule sensitivity in single cells. », *Science*, vol. 329, n° 5991, jul 2010, p. 533–538. 70

[89] YU (J.), XIAO (J.), REN (X.) *et al.*, « Probing gene expression in live cells, one protein molecule at a time. », *Science*, vol. 311, n° 5767, mar 2006, p. 1600–1603. 70

[90] ISHIZUKA (H.), HANAMURA (A.), INADA (T.) et AIBA (H.), « Mechanism of the down-regulation of cAMP receptor protein by glucose in *Escherichia coli* : role of autoregulation of the crp gene », *EMBO J.*, vol. 13, n° 13, Jul 1994, p. 3077–82. 71

[91] SNOEP (J. L.), VAN DER WEIJDEN (C. C.), ANDERSEN (H. W.) *et al.*, « DNA supercoiling in *Escherichia coli* is under tight and subtle homeostatic control, involving gene-expression and metabolic regulation of both topoisomerase I and DNA gyrase », *Eur. J. Biochem.*, vol. 269, n° 6, Mar 2002, p. 1662–9. 71

[92] WEINSTEIN-FISCHER (D.), ELGRABLY-WEISS (M.) et ALTUVIA (S.), « *Escherichia coli* response to hydrogen peroxide : a role for DNA supercoiling, topoisomerase I and Fis », *Mol. Microbiol.*, vol. 35, n° 6, Mar 2000, p. 1413–20. 71

[93] SCHNEIDER (R.), TRAVERS (A.) et MUSKHELISHVILI (G.), « The expression of the Escherichia coli fis gene is strongly dependent on the superhelical density of DNA », *Mol Microbiol*, vol. 38, n° 1, Oct 2000, p. 167–75. 71

[94] TRAVERS (A.), SCHNEIDER (R.) et MUSKHELISHVILI (G.), « DNA supercoiling and transcription in *Escherichia coli* : The FIS connection », *Biochimie*, vol. 83, n° 2, Feb 2001, p. 213–7. 71

[95] TONON (T.), EVEILLARD (D.), PRIGENT (S.) *et al.*, « Toward systems biology in brown algae to explore acclimation and adaptation to the shore environment », *OMICS*, vol. 15, n° 12, Dec 2011, p. 883–892. 72

[96] LEMOSQUET (S.), ABDOU-ARBI (O.), SIEGEL (A.) *et al.*, « A generic stoichiometric model to analyse the metabolic flexibility of the mammary gland in lactating dairy cows », dans D. SAUVANT (P. F. J. Van Milgen) et FRIGGENS (N.), éditeurs, *Modelling nutrient digestion and utilization in farm animals*. Wageningen Academic Publishers, 2010. 76

[97] OUMAROU ABDOU ARBI, JÉRÉMIE BOURDON, SOPHIE LEMOSQUET et ANNE SIEGEL, « MetaFor : a new tool to explore metabolism flexibility in complex organisms », *BMC Bioinformatics*, 2012. submitted. 76

Curriculum vitae étendu
Jérémie Bourdon

A. Etat civil

NOM Prénom : BOURDON Jérémie
Date de naissance : 25/01/1975
Adresse professionnelle :
LINA, FST de Nantes
2, rue de la Houssinière
44322 Nantes Cedex
Tél : (+33)2-51-12-58-25
Mail : Jeremie.Bourdon@univ-nantes.fr

B. ACTIVITE POUR LA PERIODE (2003-2012)

1. Publications et production scientifique

Journaux internationaux

Toward systems biology in brown algae to explore acclimation and adaptation to the shore environment, Thierry Tonon, Damien Eveillard, Sylvain Prigent, Jérémie Bourdon, Philippe Potin, Catherine Boyen, Anne Siegel*, OMICS: A Journal of Integrative Biology,* 15(12): 883-892*,* 2011

Integrating Quantitative Knowledge Into a Qualitative Gene Regulatory Network, Jérémie Bourdon, Damien Eveillard, Anne Siegel*, PLOS Computational Biology,*9(7) : 1-11, 2011

Statistical Properties of Factor Oracles, Bourdon, J. and Rusu, *Journal of Discrete Algorithms*, 9 (1), pp. 59-66, 2011.

A parallel scheme for comparing transcription factor binding sites matrices, Solenne Carat, Rémi Houlgatte and Jérémie Bourdon, Journal of Bioinformatics and Computational Biology, 8(3) pp 485-502, 2010.

Temporal constraints of a gene regulatory network: Refining a qualitative simulation, Ahmad, J.; Bourdon, J.; Eveillard, D.; Fromentin, J.; Roux, O. & Sinoquet, C., Biosystems 98(3), p. 149-159, 2009.

Combinations of cytochrome p450 gene polymorphisms enhancing the risk for sporadic colorectal cancer related to red meat consumption. Sébastien Küry, Bruno Buecher, Sébastien Robiou du Pont, Catherine Scoul, Véronique Sébille, Hélène Colman, Claire Le Houérou, Tanguy Le Neel, Jérémie Bourdon, Roger Faroux, Jean Ollivry, Bernard Lafraise, Louis-Dominique Chupin, and Stéphane Bézieau. Cancer Epidemiology Biomarkers and Prevention, 16:1460–1467, 2007.

On the Khintchine constant for centred continued fraction expansions, Jeremie Bourdon, Appl. Math. E-Notes, 7 (2007), 167-174
Statistical Properties of Similarity Score Functions, Bourdon, J. and Mancheron, A., In Proceedings of the 4th Colloquium on Mathematics and Computer Science. Algorithms, Trees, Combinatorics and Probabilities, Discrete Mathematics and Theoretical Computer Science (DMTCS), pages 129-140, (2006)

Pattern Matching Statistics on Correlated sources, Bourdon, J and Vallée, B., Proceedings of LATIN'06, LNCS 3887, pages 224-237. (2006)

Chapitres de livre

Bourdon, J. & Eveillard, D., *Probabilistic models in systems biology*. In book *Algorithms in Computational Molecular Biology: Techniques, Approaches and Applications.* Elloumi & Zomaya (eds.), Wiley Series in Bioinformatics, 2011.

A generic stoichiometric model to analyse the metabolic flexibility of the mammary gland in lactating dairy cows, Lemosquet Sophie; Abdou-Arbi Oumarou; Siegel Anne; Guinard-Flament Jocelyne; Van Milgen Jaap; Bourdon Jérémie, In Modelling nutrient digestion and utilization in farm animals, D. Sauvant, J. Van Milgen, P. Faverdin and N. Friggens eds, Wageningen Academic Publishers, 2010.

Conférences internationales

Statistical Properties of Factor Oracles, Bourdon, J. and Rusu, In proceedings of CPM 2009, LNCS 5577, 326-338, 2009 (NB : article publié dans JDA en version étendue).

Complex update strategies for Probabilistic Boolean Networks, Merle Théo; Bourdon Jérémie, In Proceedings of the 7th Workshop on Computational Systems Biology, June 2010, Luxembourg, France. TISCP, Finland, 67-71.

A statistical method for PWM clustering, Carat, S., Houlgatte, R. and Bourdon, J., In Proceedings of Moscow Conference on Computational Molecular Biology, 89-90 (2009).

A Generic Stoichiometric Model to Analyse the Metabolic Flexibility of the Mammary Gland in Lactating Dairy Cows, Lemosquet, S.; Bourdon, J.; Guinard-Flament, J.; Siegel, A. & Milgen, J. V. 7th International Workshop : "Modelling Nutrition Digestion and Utilization in Farm Animals", 2009

Conférences nationales avec comité de lecture et posters

Integrating qualitative and quantitative knowledge within a probabilistic framework: application to gene regulatory and metabolic reaction networks, Jérémie Bourdon, Damien Eveillard and Anne Siegel, Poster, CMSB 2012.

Computing the p-values of selections in genome-wide studies, Jérémie Bourdon, Solenne Carat and Mireille Régnier, Poster RECOMB 2010.

La calculette métabolique : une approche de modélisation biochimique pour explorer les contraintes au sein du métabolisme des mammifères, Arbi, O. A.; Bourdon, J.; Siegel, A.;

Milgen, J. V. & Lemosquet, S. INRA (ed.) 3èmes Journées d'animation scientifique de Département de Physiologie Animale et Systèmes d'Elevage (PHASE), 2009, p. 132

Using a probabilistic approach for integrating heterogeneous biological knowledges, Bourdon, J., Eveillard, D., Gabillard, S. & Merle, T., In Proceedings of RIAMS 2007.

Distribution des solutions du problème d'affectation multi-objectif et relation avec l'efficacité des algorithmes de résolution, Przybylski, A.; Bourdon, J. & Gandibleux, X. Proc. of ROADEF 2007, Grenoble, France, 2007

Distribution of Solutions of Multi-objective Assignement Problem and Links with the Efficiency of Solving Methods, Przybylski, A.; Bourdon, J. & Gandibleux, X. Proc. of GOR 2007, Saarbrucken, Germany, 2007

Conférences invitées

Réseaux biologiques semi-quantitatifs : dynamique et propriétés émergentese, Bourdon, J. . IV-ème école thématique Rennaise sur les systèmes complexes, Rennes, octobre 2012

Application des systèmes dynamiques : de l'arithmétique à la bioinformatique, Bourdon, J. . Ecole des jeunes chercheurs du GdR Informatique et Mathématiques, Rennes, 2012

Average-case analysis methods dedicated to the study of Biological Networks, Bourdon, J. Moscow Conference on Computational Molecular Biology (MCCMB'11), juillet 2011.

Optimisation problems and Graph Theory, Bourdon, J. . International Summer School ``Interfaces between Physics and Computer Science'', H. Meyer-Ortmanns, Bremen, 2007

Dynamical Systems and Analysis of Algorithms. Bourdon, J., conference on Optimization algorithms and quantum disordered systems, L. Cugliandolo, R. Monasson, Paris, 2004.

2. Encadrement doctoral et scientifique

J'ai participé ou participe à l'encadrement de trois thèses :

– Solenne Carat (50%, co-encadrement R. Houlgatte) Intégration de données de transcritpion et de ChIP-chip: application au réseau de transcription du récepteur d'œstrogène ESR1, soutenue en octobre 2010.
– Oumarou Abdou-Arbi (co-encadrement A. Siegel) Etude de la flexibilité métabolique : application à l'analyse du réseau métabolique de la glande mammaire de la vache, thèse démarrée en septembre 2010
– Vincent Picard (40%, co-encadrement A. Siegel) Analyse dynamique d'algorithmes et dynamique symbolique pour l'étude de modèles semi-quantitatifs en biologie des systèmes.

J'ai co-encadré 5 stages de niveau Master 2 Recherche dans la période 2007-2012:
– Théo Merle : (50%, co-encadrement R. Andonov) Comparaison de formalismes hybrides de description de réseaux de gènes.
– Oumarou Abdou-Arbi (50%, co-encadrement A. Siegel) stage en internship INRIA, Analyse du réseau métabolique de la glande mammaire de la vache.
– Solenne Carat (50%, co-encadrement R. Houlgatte) Construction et étude du réseau de transcription du récepteur d'œstrogène ESR1.
– Vincent Palmieri : (50%, co-encadrement I. Rusu) Analyse en moyenne d'un algorithme de compression d'un arbre des suffixes en oracle des suffixes.

Enfin, j'ai encadré de nombreux TER et plusieurs stages de première année de l'ENS Cachan antenne de Bretagne.

3. Rayonnement

Mes travaux de recherche portent sur le développement de méthodes probabilistes, issues de l'analyse en moyenne d'algorithmes. J'ai appliqué ces méthodes successivement à des problématiques issues de l'arithmétique, de la théorie de l'information pour me focaliser par la suite sur des problématiques bio-informatiques.
J'ai ainsi contribué, au cours de ma thèse, au développement des techniques dites de l'analyse dynamique d'algorithmes, développées autour de Brigitte Vallée. Ces techniques ont été appliquées à des études en moyenne de paramètres d'algorithmes Euclidiens.

Je me suis ensuite intéressé à des questions issues de la théorie de l'information. J'ai développé, avec Julien Clément et Brigitte Vallée (GREYC, Caen), un modèle probabiliste général, les sources dynamiques, qui découle des études de modèles dynamiques en arithmétique. Il permet de représenter très fidèlement les corrélations entre des symboles successifs d'une séquence. Ce modèle a été appliqué à des problèmes d'analyse en moyenne de paramètres de structures de données arborescentes puis il a été appliqué à des problèmes de recherche de motifs : recherche des positions d'occurrence d'une expression régulière; et recherche des occurrences de séquences de langages (motifs généralisés). J'ai aussi appliqué des méthodes probabilistes plus classiques pour résoudre trois problèmes précis : l'étude des fonctions de scores de similarité (étude qui a donné lieu au développement d'un

nouvel algorithme d'extraction de motifs); l'étude d'une structure d'index, l'oracle des facteurs; la mise au point d'une méthode de clustering de motifs.

Dans la continuité de ces dernières études en théorie de l'information, je me suis intéressé à des problèmes de biologie des systèmes. C'est devenu une thématique importante dans mes recherches et je collabore actuellement dans ce domaine tant avec des équipes de biologistes (Institut du Thorax, CHU de Nantes ; Station Biologique de Roscoff ; Université de Santiago du Chili ;…) que des équipes d'informatitiens (EPI SYMBIOSE, IRISA ; MoVeS, IRCCyN ; Freie Universität Berlin,…). Je me suis intéressé à l'étude des réseaux en y abordant des aspects dynamiques, constatant que les trajectoires peuvent être vues comme des mots particuliers émis par une source complexe. Les méthodes que j'ai développées permettent de fournir des informations intéressantes et originales sur la source qui a émis ces trajectoires, donc sur le système. Le premier travail a permis de mettre au point une méthode probabiliste de réduction de modèle pour rendre possible l'application de techniques de vérification formelles sur des automates hybrides qui prennent en compte des aspects chronométriques dans les modèles. Le second travail a consisté en une intégration quantitative de l'échelle de représentation protéique dans les réseaux de régulation de gènes.

De manière plus marginale, j'ai collaboré à une étude statistique des SNPs impliqués dans le cancer du colon (Collaboration avec le LEPA, CHU de Nantes). Je me suis aussi penché sur des problèmes de tests multiples en développant une méthode, basée sur des calculs de fonctions génératrices, pour calculer de manière exacte le nombre de faux positifs impliqué par une certaine sélection lors d'un choix multiple, c'est une question restée ouverte depuis de nombreuses années et qui trouve de nombreuses applications (collaboration Equipe-projet AMIB, INRIA Saclay). Je me suis également intéressé à l'étude théorique du coût des algorithmes d'optimisation multi-objectifs en fonction de la distribution des instances (collaboration équipe ROOM du LINA).

Enfin, profitant de deux années de délégation INRIA dans l'équipe projet SYMBIOSE, j'ai développé des collaborations sur trois sujets en biologie des systèmes et en analyse en moyenne d'algorithmes. Avec Anne Siegel (Equipe-projet SYMBIOSE, INRIA Rennes-Bretagne-Atlantique), nous avons débuté une collaboration prometteuse avec l'INRA Saint-Gilles sur l'étude du réseau métabolique de la glande mammaire de la vache. Cette collaboration a été fructueuse, grâce notamment à un co-encadrement de stage en internship (sur ce sujet, l'étudiant a obtenu le prix Ibni décerné par la Société Mathématique de France). Cet étudiant poursuit actuellement une thèse sur ce sujet à Rennes, je participe activement à son encadrement. Avec Anne Siegel, nous avons mis au point un modèle original, le graphe d'évènements de transition qui est une abstraction (linéaire) de la dynamique (exponentielle) d'un système. L'intérêt de ce modèle est qu'il permet d'intégrer des données quantitatives (par exemple des évolutions de concentration de protéines) dans des modèles purement qualitatifs de description du vivant. Ce modèle nous a permis de modéliser le changement de phase de la bactérie *E. Coli* en privation de carbone, un système vivant servant de « benchmark » pour de nombreuses méthodes de modélisation. Nous appliquons les méthodes développées à un modèle plus ambitieux, en collaboration avec des chercheurs biologistes de l'UMR Mer et Santé, Station Biologique de Roscoff, UPMC. Cette collaboration est financée par un projet PEPS CNRS (QuantOursin), 2010-2012. En parallèle, nous travaillons sur une adaptation

de ces méthodes pour étudier l'assimilation du cuivre par des bactéries avec une application à l'exploitation minière au Chili, en collaboration avec l'université de Santiago du Chili (Center of Mathematical Modeling, UMI 2807).
Enfin, avec François Coste (Equipe-projet SYMBIOSE, INRIA Rennes-Bretagne-Atlantique), nous avons étudié en moyenne les scores qui régissent certains algorithmes d'inférence grammaticale. Nous avons co-encadré un étudiant de l'ENS Cachan et rédigeons un article de journal sur ce sujet.

Projets financés sur la période.
J'ai coordonné le projet inter-laboratoires BioAtlanSTIC "Vérification formelle de propriétés pour un système dynamique stochastique" entre les équipes MoVeS et ComBi en 2007-2009. Ces projets sont distribués par la fédération de recherche AtlanSTIC (FR CNRS 2819) pour favoriser les interactions autour d'équipes informatiques Nantaises.
En outre, dans la période 2006-2009, j'ai participé à l'ANR Blanche SADA. J'ai aussi participé au projet régional Bio-Informatique Ligérienne (2006-2010, porteur Rémi Houlgatte, INSERM U915) dont le but était de fédérer les acteurs de bio-informatique en Pays de Loire par le financement de thèses notamment. Je participe actuellement aux ANR blanches LAREDA (2008-2011, porteur : B. Vallée), sur un sujet qui dans la suite de mes travaux de thèse et GUEPARD (2010-2013, porteur : P. Perny, LIP6), analyse en moyenne des problèmes d'optimisation multi-objectifs.
Je participe très activement au projet PEPS CNRS QuantOursin (2010-2012, porteur A. Siegel, IRISA), modélisation quantitative de l'initiation de la traduction chez l'oursin. Je suis responsable de partenaire pour l'ANR Blanche BioTempo (2011-2014, porteur A. Siegel, IRISA).
Enfin, je participe, en tant que sous-contractant de l'IRISA au projet d'investissement d'avenir « Biotechnologies et bioressources » IDEALG, projet financé sur 10 ans, dont le but est notamment d'étudier les phénomènes d'adaptation de certaines algues à des modifications de l'environnement.
Il est à noter que plusieurs de ces projets découlent des collaborations entamées au cours de ma délégation.

4. Responsabilités scientifiques

- Représentant du domaine « bio-informatique » au conseil scientifique de Biogenouest depuis 2004. Biogenouest est la génopole de bioinformatique de Bretagne et Pays de Loire (environ 60 laboratoires en font partie).
- Editeur pour la revue PloS One.
- Membre du comité de programme de JOBIM 2012
- Nombreuses reviews pour divers journaux et conférences en bioinformatique, en combinatoire et en analyse d'algorithmes. Uniquement sur la période 2011-2012 : Bioinformatics, Annals of combinatorics, Discrete Mathematics, CMSB, Signal Processing, Analysis of Algorithms, …
- Expertises de projets ANR et NWO (agence néerlandaise de moyens),
- Membre du comité de sélection de l'université de Paris-Sud 2010, dans le vivier externe depuis.
- Chargé de mission "Budget et Finances" du LINA 2005-2008,
- Organisation de la conférence JOBIM 2009, à Nantes.
- Organisation de trois workshop sur le thème de la "Modélisation dynamique des réseaux biologiques" en 2008 à Lille,en 2009 à Nantes et en 2010 à Montpellier.

- Organisation de la journée des doctorants JDOC 2011 de l'école doctorale ED STIM 503
- Titulaire de la Prime d'Excellence Scientifique depuis 2011.

C. ACTIVITE D'ENSEIGNEMENT

Titulaire d'un poste d'enseignant-chercheur au département d'informatique de Nantes, j'y exerce une activité à temps plein. Je suis ainsi intervenu dans de nombreux enseignements que je ne détaillerai pas ici. Je me focaliserai sur quelques enseignements dont j'ai eu la responsabilité, dont j'ai défini le contenu et produit les supports. Ces enseignements sont de deux types : des enseignements des bases de l'informatique et des enseignements plus spécialisés.

Modules d'introduction à l'informatique

- *Création de sites web* (L1) : module consacré aux techniques de développement de sites web et apprentissage des langages associés (html, css, javascript et php). Responsabilité de ce cours pendant 5 ans.
- *Algorithmique et programmation 1* (L1) : premier module d'informatique pour les étudiants de licence, bases de l'algorithmique. Mise au point d'un outil pédagogique : une interface de programmation en javascript intégrant diverses bibliothèques et utilisée dans le cadre de ce cours[1]. J'ai la responsabilité de ce cours depuis 3 ans ainsi que la responsabilité de la première année de licence d'informatique.

Modules spécialisés

- *Algorithmique numérique* (L3) : module consacré aux méthodes de calcul numérique en général (algorithmes de resolution, d'interpolation,…). Responsabilité de ce cours pendant 2 ans.
- *Probabilités et statistiques* (L3 Miage) : module d'introduction aux bases des probabilités et des tests statistiques avec une application aux problèmes en gestion. Responsabilité de ce cours pendant 2 ans.
- *Probabilités et statistiques pour la bioinformatique* (M2 BioInfo) : Présentation d'outils de probabilités et statistiques et présentation d'applications en bioinformatique. Responsabilité de ce cours pendant 4 ans.
- *Cryptographie* (M1 informatique) : module présentant des éléments de cryptographie (cryptosystèmes, générateurs pseudo-aléatoires) et de sécurité (protocoles de partage de secret, d'échange de clés). Responsabilité de ce cours pendant 4 ans.

Je suis aussi intervenu lors de trois écoles de jeunes chercheurs :

- IV-ème école thématique Rennaise sur les systèmes complexes, Rennes, octobre 2012, intitulé du cours *Réseaux biologiques semi-quantitatifs : dynamique et propriétés émergentes*.
- Ecole des jeunes chercheurs du GdR Informatique et Mathématiques, Rennes, 2012, intitulé du cours *Application des systèmes dynamiques : de l'arithmétique à la bioinformatique*.
- International Summer School « Interfaces between Physics and Computer Science», intitulé du cours *Optimisation problems and Graph Theory.*

[1] https://dl.dropbox.com/u/1961350/AlgoScript.html

l'université nantes angers le mans
PÔLE DE RECHERCHE ET D'ENSEIGNEMENT SUPÉRIEUR

UNIVERSITÉ DE NANTES

Habilitation à Diriger des Recherches

Jérémie Bourdon

Sources probabilistes

des séquences aux systèmes

Résumé

Ce mémoire est un recueil et une synthèse de plusieurs études en analyse en moyenne d'algorithmes et en bioinformatique. Il y est présenté des travaux allant de l'étude de problèmes sur les séquences à l'étude de systèmes biologiques, en gardant un fil conducteur fort : quelles que soient les applications, l'objet d'étude central est une source probabiliste qui produit des mots. Ces travaux trouvent des applications en bioinformatique qui se concrétisent par la mise au point d'algorithmes dédiés de recherche de motifs et la définition de tests statistiques et en biologie des systèmes biologiques avec des développements qui ont été appliqués, en collaboration étroite avec des équipes de biologistes, à des modèles biologiques réels.

Mots clés

systèmes dynamiques, séquences biologiques, Biologie des systèmes.

Abstract

This monograph synthesizes several studies in average-case analysis of algorithms and in computational biology. Several works, spanning from the study of problems on sequences to systems biology, are presented. A unifying thread is followed : whatever the applications are, probabilistic sources of words plays the central role. These works find applications both in biological sequence analysis with a focus on the improvement of pattern matching algorithms and in the design of precise statistical tests, and in systems biology where strong collaborations with biologists permit to apply the developed methods to real living systems.

Key Words

Dynamical systems, Biological sequences, Systems biology.